Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 100 (2015) 1099 – 1107

Procedia Engineering

www.elsevier.com/locate/procedia

25th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM 2014

Platform's Architecture for Colorectal Cancer Research in the Cloud

Monika Simjanoska*, Marjan Gusev, Sasko Ristov

University Ss. Cyril and Methodius,Faculty of Computer Science and Engineering, Rugjer Boskovikj, 16, 1000 Skopje, Macedonia

Abstract

The Colorectal Cancer (CRC) is one of the most frequent causes of death from cancer in the developed regions. In this paper we address the necessity to develop tools by which the oncologists will easily identify molecular targets for treatment and perform immediate diagnosis. Considering the increase of available microarray experiments and the DNA chips' potentiality for extracting knowledge from the genes' expressions, we present the design of a new platform for CRC gene expression analysis whose usage target are both the researchers and the medical persons. The platform is set on predefined and original statistical methodologies for accurate inference in CRC, biomarkers extraction, CRC stage prognosis, etc. In order to avoid the limitations of the local infrastructure, the proposed architecture takes the advantage of the cloud's speed and flexibility to perform large-scale computations and to be easily upgradable with new tools. The platform we present is equivalent to the Software as a Service level of abstraction from an end-user point of view. From a developer's point of view, we use the Platform as a Service level of abstraction, which means we control virtualized instances that are already offered by the Cloud Service Provider. Those instances are set on the Infrastructure as a Service level of the cloud, its storing and processing capacity. Mainly, the platform is comprised of four distinctive modules: User Interface – interface between the user, the software and the biological databases; Machine Learning – the core where the data is processed; Distributed Processing – splits and spreads the problem through the available instances; and Resources Controlling module – determines the best combination of resources in terms of performance and cost depending on the problem size. A detailed description of each module, as well as the latest concerns of cloud's interoperability and portability is discussed in the paper.

© 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Peer-review under responsibility of DAAAM International Vienna

Keywords: Cloud Computing; Colorectal Cancer; SaaS; Microarray; Gene Expression

* Corresponding author. Tel.: +38976472195
E-mail address: m.simjanoska@gmail.com

1877-7058 © 2015 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

Peer-review under responsibility of DAAAM International Vienna
doi:10.1016/j.proeng.2015.01.472

1. Introduction

Cancer is one of the most challenging fields for research. Due to its individuality, the molecular profiling is the most recently used approach for characterizing the tumor's genetics. The gene expression microarray technology gives an insight of how a cell responds to changed conditions and has the potential to revolutionize the cancer's treatment and diagnosis by visualizing the expression of thousands of genes simultaneously.

CRC is one of the most frequent causes of death from cancer in the developed regions worldwide [1]. In this paper we address the necessity to develop tools by which the oncologists and the researchers will easily identify molecular targets for treatment, discover relevant biomarkers, determine the individual recurrence risks, perform immediate diagnosis and determine the stage of CRC progression. The development of the tools depends on strong statistical background, which means analysis of thousands of genes from the many widely used microarray technologies as Affymetrix, Illumina, Agilent, etc. All the arrays we mentioned have made complete sequencing affordable to small laboratories. However, acquiring the gene expression is only the first step that must be followed by large-scale computational analysis to process the data. Consequently, the laboratories are forced to invest in computer hardware and skilled staff to perform data analysis [2].

Among the many challenges that we confront in this paper is the need of retrieving microarray data, combining different sources of gene expression, different microarray platforms, allowing data management, developing methodologies for data transformation, strategies for knowledge extraction and modelling, and machine learning techniques for applying the models. To respond to these tasks we need a set of different tools, which we integrated in a software platform. Gentleman et al. [3] clearly defined the primary motivations for an open-source computing environment to be:

- Transparency - Since complex steps are involved in the conversion of information from low-level information structures to statistical databases of expression measures.
- Pursuit of reproducibility - All the experiments should be done by obeying the established protocols and should be accompanied by the source code and the data on which the analysis is based.
- Efficiency of development - Since an open-source environment with good documentation is beneficial for investigators who want to extend it with new functionalities, for those who can use it for learning purposes and also for recruitment and training of future generations of scientists.

Therefore, in this paper we present the design and architecture of a new platform based on open source computing environment for CRC gene expression data analysis that eliminates the hardware and skills issues. The platform is designed to be host for tools that rely on original methodologies developed for CRC tissues analysis whose expression is obtained from different types of microarray technologies. As the number of available microarray experiments continues to rise, the methods are becoming more and more compute-intensive and the need for resources to perform parallel computations is likely to grow. If this trend continues, the local infrastructure will become unmanageable in terms of capacity and cost. Therefore, considering the large number of studies that claim the cloud computing environments are both performance and cost-effective solution for bioinformatics applications, we designed our platform to take the advantage of the cloud's speed and flexibility to perform large-scale computations and to be easily upgradable with new tools.

The rest of the paper is organized as follows. In Section 2 we present the latest platforms and tools developed for analysis of microarray data. In Section 3 we present the architecture and the design of the cloud platform whose aim is to host tools specialized for the analysis of CRC gene expression data and serves to both the researchers and the medical persons for the purpose of investigating new methodologies and use them conditionally. The interoperability and portability challenges of SaaS applications are discussed in Section 4. In the final Section 5 we present summary of the proposed architecture and the benefits of introducing the platform.

2. Related work

In this section we present some of the promising tools developed in the past decade whose purpose is to ease the analysis of microarray experiments.

ArrayMining is software created by Glaab at al. [4]. Actually, it is a web application that is able to combine several high-dimensional data sets and algorithms while performing feature selection, clustering, prediction, gene set analysis and cross-study normalization. They put an accent on the cross-study feature as an advantage in comparison with the existing tools that mainly analyze single experiments.

GEPAS is a web-based package for gene expression analysis that is continuously updated in the past years [5]. The newest release is defined as integrative platform with new features and improvements considering the most common microarray formats, normalization, gene selection, class prediction, clustering, time-series analysis, stratification analysis, association, functional enrichment and gene set enrichment analysis with functional terms, text-mining, derived bioentities and protein-protein interaction analysis. For future improvement, they address several features: normalization for Illumina arrays and interface to ArrayExpress [6] and Gene Expression Omnibus [7] databases, which we have already integrated in our proposed platform.

Another open source and web-based suite is Asterias [7] that focuses on the analysis of gene expression and aCGH data. Beside the implementation of the validated statistical techniques, the tools are parallelized to take benefit of multicore CPUs and computing clusters. The authors' efforts are mainly focused on making Asterias easy to install and deploy on both laptops and clusters. Our design overcomes this problem by utilizing the advantages from cloud computing and thus make everything accessible by using only a web browser, but still with highly preserved performance when solving problems of different scale.

EzArray developed by Zhu et al. [8] is a system for management and analysis of Affymetrix gene expression data. It also provides interface to GEO database with the purpose to re-analyzing previously published microarray data. However, the tool is limited only to Affymetrix technology and cannot be used for other types and format of microarray data. On the other hand there is a web tool that is specialized only for Agilent microarray experiments [9]. In our platform we intent to introduce all widely used technologies with along with the mostly used subtypes of chips.

Considering the cloud computing paradigm whose infinitely scalable infrastructure is a great opportunity for high-dimensional analysis, there is still deficit in cloud applications specialized for microarray analysis. Yang et al. [10] developed BioVLAB-Microarray that utilizes remote high-performance computing resources and provides flexible and configurable virtual environment for microarray gene expression analysis whose targets are small research labs that lack equipment and expertise. Even though they offer variety of methods, they are not combined in well-established methodologies for analyzing gene expression obtained from different types of arrays. Instead, the combination of methods for discovering biomarkers, clustering, etc., depends on the user itself. The software has also a newer extension BioVLAB-MMIA - an environment for the integrated analysis of microRNA and mRNA expression data [11].

YunBe is another specially designed algorithm for biomarker identification in the cloud [12]. It is written in Java and uses MapReduce framework to parallelize the analysis. The developers compared the execution speeds of the program on 1, 2, 4 and 8 Amazon's EC2 m1.large instances to 2, 4, 8 and 16 cluster cores. Comparisons have also been made with a desktop program. The results showed that in comparison to a desktop implementation, YunBe significantly improves execution times.

There are, however, platforms that support the development of bioinformatics infrastructures on the cloud.

CloVR [13] relies on virtual machines (VMs) and compute clouds to provide improved access to bioinformatics workflows and distributed computing resources. It provides a single VM containing pre-configured and automated pipelines, suitable for easy installation on the desktop and with cloud support for increased analysis throughput. The authors in their paper evaluate the features of the CloVR architecture as portability across different local operating systems and remote cloud computing platforms, support for elastic provisioning of local and cloud resources, scalability of the architecture and use of local data storage on the cloud.

Cloud BioLinux [14] is a publicly accessible VM that enables scientists to quickly provision on-demand infrastructures for high-performance bioinformatics computing using cloud platforms. Users have instant access to a range of pre-configured command line and graphical software applications, including a full-featured desktop interface, documentation and over 135 bioinformatics packages for applications including sequence alignment, clustering, assembly, display, editing, and phylogeny. Besides the Amazon EC2 cloud, they have started instances of Cloud BioLinux on a private Eucalyptus cloud.

The authors in [15] present a cloud resource management system, CloudMan, which makes it possible for individual researchers to compose and control an arbitrarily sized compute cluster on Amazon's EC2 cloud

infrastructure. They have provided a mechanism for streamlining a tool installation process. Their solution makes it possible to create a completely configured compute cluster ready to perform analysis in less than five minutes.

GenomeSpace [16] is a cloud-based interoperability framework to support integrative genomics analysis through an easy-to-use Web interface. GenomeSpace provides access to a diverse range of bioinformatics tools, but does not perform any analyses itself. The analyses are done within the member tools wherever they live (desktop, web service, cloud, server, etc.). It acts as a data highway automatically reformatting data as required when results move from the output of one tool to input for the next. GenomeSpace hosts variety of tools and data sources that provide a wide spectrum of genomic analysis and bioinformatics capabilities, as: Cytoscape, Galaxy, GenePattern, Genomica, IGV, InSilico DB, Cistrome, geWorkbench, ArrayExpress, Gitools, ISAcreator, Synapse, etc.

3. Platform's architecture and design

Bioinformatics is of interdisciplinary nature, connects the advances of biology (medicine) and information technology, and therefore needs to handle a vast quantity of biological data generated by high-throughput experimental technologies. Considering the high-demanding nature of the experiments, bioinformatics is experiencing a new leap-forward from in-house computing infrastructure into utility-supplied cloud computing delivered over the Internet [17]. According to the NIST definition, cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [18].

To utilize the "infinitely" scalable infrastructure, we propose architecture depicted in Fig. 1. The platform we present is equivalent to the Software as a Service (SaaS) level of abstraction from an end-user point of view. The installation of the software (platform) is managed by a Software Service Provider (SSP) and its users can access the tools hosted on the platform anytime, and can store their experimental data safely in the infrastructure. From a developer's point of view, we use the Platform as a Service (PaaS) level of abstraction, which means we control virtualized instances that are already offered by the Cloud Service Provider (CSP). Those instances are set on the Infrastructure as a Service (IaaS) level of the cloud, its storing and processing capacity.

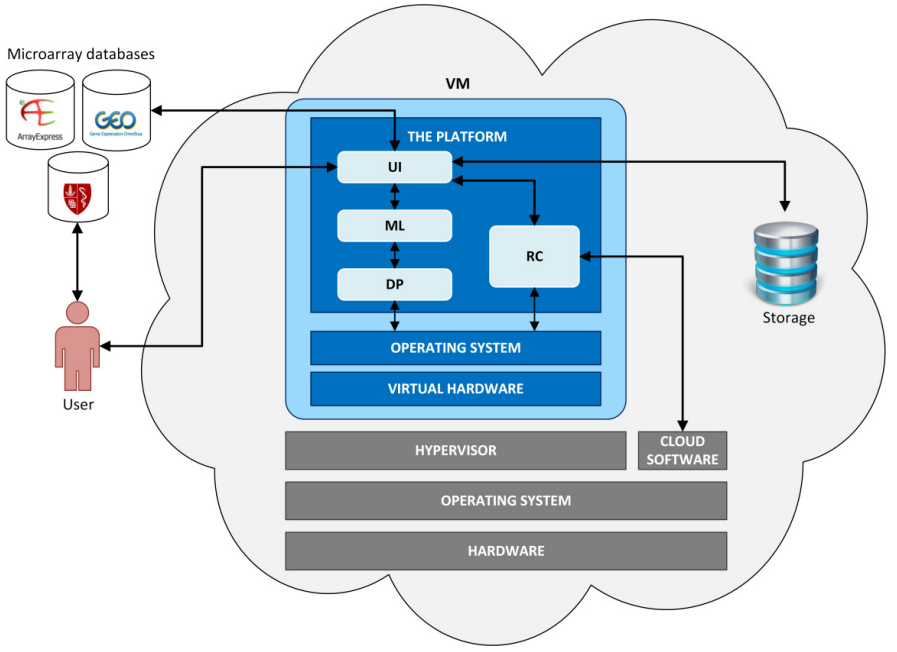

Fig. 1. Platform's architecture.

The CRC gene expression analysis platform is comprised of four different modules: User Interface (UI) Module, Machine Learning (ML) Module, Distributed Processing (DP) Module and Resources Controlling (RC) Module. Each module is discussed in the following sections.

3.1. UI module

The UI Module is on top of the other modules of the platform and is responsible to provide the user with all the interfaces necessary for appropriate usage of the tools' functionalities. The platform's tools are specially designed for the analysis of CRC gene expression values and therefore, it will provide interfaces for direct connection with some of the most frequently used biological databases that store microarray data as: NCBI's Gene Expression Omnibus, EMBL's ArrayExpress, Stanford Microarray Database, etc. For a confident and secure research, the results from the analyses will not be available in public, instead each user will be required to provide its own login information and thus a new storage will be created for saving user's data and results.

3.2. ML module

This module is the core of the platform. It is a package of methodologies which are sums of functions written in the programming language R. The methodologies are set on series of analyses and are developed as original procedures, each referring to a different problem or technology used for the experiments. In order to cover most of the assumed requirements, we continuously expand our research and by now we have worked on and proposed solutions for the following CRC gene expression challenges:

- Significant genes detection – This problem refers to the elimination of the genes that are not important for the CRC and therefore, reduces the set from thousands to hundreds of genes. In order to define an appropriate methodology, the researchers must take into account both the technology used for measuring the expression of the genes in the tissues, and also the statistical and biological significance of the gene expression levels. In our recent research we analyzed gene expression data obtained from two widely used technologies, Illumina [19] and Affymetrix [20]. The development of the distinct methodologies is confirmed by the comparison reported in our research [21], where we proved that the each methodology is platform-depended. The existing methodology will be upgraded with the latest computation-intensive method we proposed [22] and which we aim to parallelize in the near future. Based on this experience we will continue our future research to cover more Illumina and Affymetrix subtypes of chips, and also to create biomarkers detection procedures for as many new technologies as possible.
- Tissue diagnosis - This type of analysis is of classification nature. Usually, the technologies of our interest are dual-channel, which means they provide the opportunity to observe gene expression of carcinogenic and healthy tissues in parallel. We took the advantage and developed original distinct methodologies for each technology [19, 20] whose aim is to determine whether the tissue suffers CRC or not. The diagnosis is based on strong statistical background where the expression of the significant genes (biomarkers) is modeled and used in the Bayes' theorem. Since the biomarkers discovered from the previous methodologies showed satisfying discrimination capabilities when used in other classification techniques as Support Vector Machines, Decision Trees, etc., the proposed CRC platform in the cloud will also integrate those techniques and offer them as optional classification tools.
- CRC stage determination - This is very sensitive area of research for the CRC problem due to the problematic distinction between the critical stages reported in the literature - stage I with stage IV and stage II with stage III. This problem was discussed in our previous research [23] where we proposed multi-classification solution with carefully defined preprocessing procedure for the different CRC stages. In order to improve the classifier's precision, we continued the research to integrate Gene Ontology (GO) as an additional indicator for the biomarkers' significance [24]. Determining the actual stage of the cancer is of great importance for early prognosis and appropriate treatment of the disease.

All the methodologies reported in the paper are continuously upgraded with novel approaches and discoveries. Some of them can also be used for more detail CRC investigation. For example, the biomarkers obtained from

significant genes detection methodologies were analyzed with GO tools to discover the functional groups they are associated with [25]. Investigating the genes that control the colorectal carcinogenic tissue development is of great importance to the biomedical researchers and also a good indicator for the validity of the methodologies we created. However, there are still CRC problems that remain uncovered and will be a target of our future work.

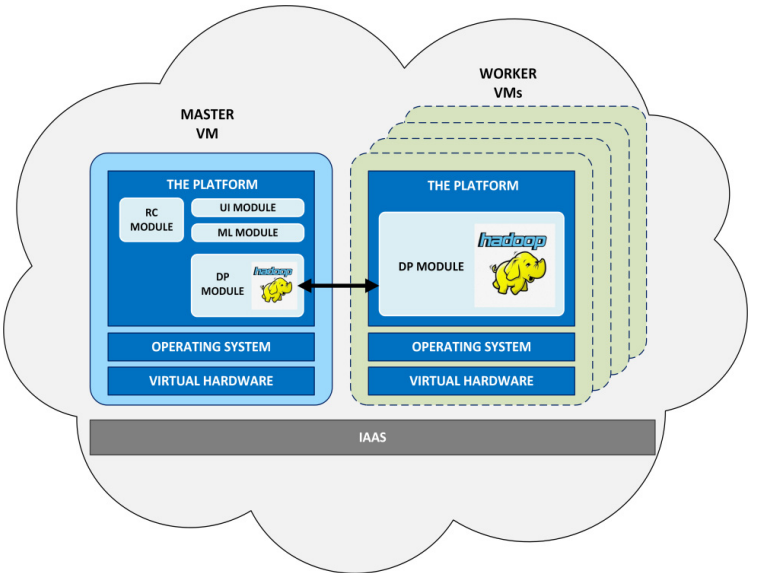

Fig. 2. Platform's initial setting in the cloud.

3.3. DP module

This module intermediates between the ML module of the application and the virtualized resources of the VM instances. In order to perform parallel analysis we assume that Apache's Hadoop [26] software framework is installed on top of the operating system (OS). Hadoop is an open source implementation of the MapReduce algorithm that divides the problem into smaller subproblems, distributes them among the available processors, collects the results and then combine them into one output. The framework is designed to scale up to thousands of machines, each offering local computation and storage.

Considering bioinformatics applications, Hadoop is frequently used for large scale analysis. FX is a tool for RNA-Seq analysis that runs in Hadoop systems as well as in the Amazon cloud [27]. Eoulsan is a scalable framework that is also based on the Hadoop implementation of the MapReduce algorithm dedicated to high-throughput sequencing data analysis [28]. Gunarathne et al. [29] have demonstrated how Apache Hadoop Map Reduce distributed parallel computing framework offers a simple programming model and a convenient user interfaces with little overhead to scientific computation applications. Taylor [30] in his research gives an overview of the current usage of Hadoop within the bioinformatics community. As a conclusion the author states that not only the scalability is important, but also the simplicity of integrating and analyzing various large data sources into one data warehouse under Hadoop.

Taking into account the various confirmations of the Hadoop convenience in bioinformatics, we set the initial setting of our proposed platform to be organized as depicted in Fig. 2. In order to have an appropriate implementation of the Hadoop's master/slave architecture, there must be at least two VM instances. Therefore, we propose a setting where the CRC platform with all its modules is set on a static VM instance, referred to as *Master VM*, and a copy of the platform that contains only the DP module is set on another VM instance, which we call *Worker VM*. The Worker doesn't host the whole platform's modules since it only provides computing resources to the *Master VM*. The number of workers depends on the problem size and is controlled by the intelligent algorithm integrated in the RC module explained in the following section.

3.4. RC module

The existing load balancing solutions are used to balance the requests between two or more instances of client's applications in a way that they can be provisioned automatically without requiring changes to the network or its configuration [31]. Our proposed platform for CRC gene expression analysis is not expected to handle large number of concurrent users, instead the problem is to handle the possibly high-demanding computation demands of a particular user. Therefore, the existing load balancing solutions are no more beneficial and a new module for controlling the resources is proposed - the *RC Module*.

The RC module's aim is to observe the available computing resources and to derive intelligent decisions of the most performance and cost-effective configuration, which means it takes the responsibility to start or shut down *Worker VM* instances. The module is integrated in the CRC platform and is hosted on the *Master VM*. As depicted in Fig. 1, the resource controller exchanges information with the user interface, the cloud software and the operating system. Therefore, the whole decisive algorithm relies on two inputs:

- Problem size - Instead of a continuous measurement of the resources utilization, the problem size is calculated and obtained from the UI module as soon as the user defines the experimental settings. This approach saves time and makes the algorithm to be of a predictive nature.
- CPU Utilization - This parameter is also calculated in advance, before the user starts with the analysis. The information is obtained from the web services that are hosted on each *Worker VM* instance. Each web service captures the CPU utilization of its host and returns the information to the *Master VM*. However, this source of information is possible only after the RC module communicates with the cloud software and collects all the *Worker VM* instances' IP addresses.

In order to achieve both performance and cost-effective configuration in the cloud, we must develop a methodology based on machine learning analysis of various user loads in different cloud environments. This methodology will be the core of the RC module and a key for determining the most appropriate configuration for a given problem size. Recently, we did some research whose results will be used while designing the RC module.

To inspect the impact of variable server load on the performance and the cost of the rented resources we have already performed analysis of a memory demanding and computation intensive web services which we hosted in the basic VM instances currently offered by the CSPs. The results from the experiments showed that some conclusions can be derived in terms of lowest cost with satisfying performance gain; however, the performance gain still depends on the problem size, even though generally it stays positive [32].

In another research we made an effort to provide a realistic modelling of previously measured response time and CPU utilization of VM instances with different number of cores. Hereupon, we proposed a well-defined preprocessing methodology to produce data applicable for machine learning analysis. The problem analyzed was whether our proposed procedure will be able to map the response time and the CPU values into the right cloud setting involved in the testing. The results from the classification showed high accuracy and also high ability when proposing a configuration that will provide the user with maximum performance but still to be a cost-effective solution [33].

4. Discussion

In this section we discuss the importance of interoperability and portability challenges when developing our SaaS platform in the cloud.

In SaaS level, interoperability refers to the ability of SaaS systems on one cloud provider to communicate with SaaS systems on another cloud provider [34]. Interoperability can be reached when two systems use the same cloud API which supports the deployment, management and monitoring of virtual workloads like VMs [35].

LISI (Levels of Information System Interoperability) is a widely recognized model for system-of-systems interoperability. It focuses on system-to-system information exchanges in terms of procedures, applications, infrastructure and data. However, the LISI model doesn't rate the cloud-to-cloud interoperability. Dowell et al. [36] discuss some challenges in making cloud interoperable and propose Cloud-to-Cloud-Interoperability model. A question of our interest is: "Can we deploy existing virtual images on another CSP without any modifications?". In

order to achieve portability, the authors state the necessity of open standards for VMs and cloud-to-cloud application interfaces - the cloud APIs. Petcu et al. [37] define two types of cloud APIs: cloud provider's and cross platforms. They state that from a cross platform API, the developers expect a unified and standardized API regardless the difference between the cloud providers. However, this issue is still not fully covered in the literature. Instead there are many theoretical models and surveys that focus on the problems of interoperability and portability in cloud computing [38, 39].

Conclusion

Considering the seriousness of the CRC incidence and prevalence, we propose a platform for analysis of gene expression data obtained from the popular microarray technologies that are widely used for cancer research. The platform is host for tools that rely on original methodologies for identification of molecular targets for treatment, discovering relevant biomarkers, determination of the individual recurrence risks, performing immediate diagnosis and prediction of the stage of CRC progression. The reason for proposing the platform is to ease the further investigation of the CRC by integrating different tools that cover all the problems discussed above. Such tools are either missing, or require too much expertise to be used.

The architecture that we propose will utilize the highly scalable infrastructure offered by the CSPs. The platform is comprised of four different modules:

- a module for providing user-friendly interface;
- a module for doing the intelligent analysis;
- a module for distributed and parallel computation of the complex demands and
- a module for managing the resources in the cloud.

All the modules discussed in the paper are based on strong statistical background and on experience from our previous research. Some of the methodologies we discuss are already proved to be relevant and can be used by the experts.

Our future work will aim to overcome the platform's portability and interoperability issues in the cloud. Furthermore, we will continue our research towards improving the existing and expanding the platform with new CRC specific tools. All the proposed methodologies will be automatically updated with new knowledge as soon as new microarray experiments are available in the databases linked to the software platform. That is how the users will be provided with the latest information of CRC and thus the relevance of their results will be improved.

References

[1] I. A. for research on cancer et al., The globocan project: cancer incidence and mortality worldwide in 2008, 2010.
[2] K. Krampis, T. Booth, B. Chapman, B. Tiwari, M. Bicak, D. Field, and K. E. Nelson, Cloud biolinux: pre-configured and on-demand bioinformatics computing for the genomics community, BMC bioinformatics, vol. 13, no. 1, p. 42, 2012.
[3] R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry et al., Bioconductor: open software development for computational biology and bioinformatics, Genome biology, vol. 5, no. 10, p. R80, 2004.
[4] E. Glaab, J. M. Garibaldi, and N. Krasnogor, Arraymining: a modular web-application for microarray analysis combining ensemble and consensus methods with cross-study normalization, BMC bioinformatics, vol. 10, no. 1, p. 358, 2009.
[5] J. T´arraga, I. Medina, J. Carbonell, J. Huerta-Cepas, P. Minguez, E. Alloza, F. Al-Shahrour, S. Vegas-Azc´arate, S. Goetz, P. Escobar et al., Gepas, a web-based tool for microarray data analysis and interpretation, Nucleic acids research, vol. 36, no. suppl 2, pp. W308–W314, 2008.
[6] G. Rustici, N. Kolesnikov, M. Brandizi, T. Burdett, M. Dylag, I. Emam, A. Farne, E. Hastings, J. Ison, M. Keays et al., Arrayexpress update trends in database growth and links to data analysis tools, Nucleic acids research, vol. 41, no. D1, pp. D987–D990, 2013.
[7] R. Edgar, M. Domrachev, and A. E. Lash, Gene expression omnibus: Ncbi gene expression and hybridization array data repository, Nucleic acids research, vol. 30, no. 1, pp. 207–210, 2002.
[8] R. D´ıaz-Uriarte, A. Alib´es, E. R. Morrissey, O. M. Rueda, M. L. Neves et al., Asterias: integrated analysis of expression and acgh data using an open-source, web-based, parallelized software suite, Nucleic acids research, vol. 35, no. suppl 2, pp. W75–W80, 2007.
[9] Y. Zhu, Y. Zhu, and W. Xu, Ezarray: a web-based highly automated affymetrix expression array data management and analysis system, BMC bioinformatics, vol. 9, no. 1, p. 46, 2008.
[10] A. L. Vollrath, A. A. Smith, M. Craven, and C. A. Bradfield, Edge3: A web-based solution for management and analysis of agilent two color microarray experiments, BMC bioinformatics, vol. 10, no. 1, p. 280, 2009.

[11] Y. Yang, J. Y. Choi, K. Choi, M. Pierce, D. Gannon, and S. Kim, Biovlab-microarray: Microarray data analysis in virtual environment, in: eScience, 2008. eScience'08. IEEE Fourth International Conference on. IEEE, 2008, pp. 159–165.
[12] H. Lee, Y. Yang, H. Chae, S. Nam, D. Choi, P. Tangchaisin, C. Herath, S. Marru, K. P. Nephew, and S. Kim, Biovlab-mmia: a reconfigurable cloud computing environment for microrna and mrna integrated analysis, in: Bioinformatics and Biomedicine (BIBM), 2011 IEEE International Conference on. IEEE, 2011, pp. 494–499.
[13] L. Zhang, S. Gu, Y. Liu, B. Wang, and F. Azuaje, Gene set analysis in the cloud, Bioinformatics, vol. 28, no. 2, pp. 294–295, 2012.
[14] S. V. Angiuoli, M. Matalka, A. Gussman, K. Galens, M. Vangala, D. R. Riley, C. Arze, J. R. White, O. White, and W. F. Fricke, Clovr: A virtual machine for automated and portable sequence analysis from the desktop using cloud computing, BMC bioinformatics, vol. 12, no. 1, p. 356, 2011.
[15] E. Afgan, D. Baker, N. Coraor, B. Chapman, A. Nekrutenko, and J. Taylor, Galaxy cloudman: delivering cloud compute clusters, BMC bioinformatics, vol. 11, no. Suppl 12, p. S4, 2010.
[16] [Online]. Available: http://www.genomespace.org
[17] L. Dai, X. Gao, Y. Guo, J. Xiao, Z. Zhang et al., Bioinformatics clouds for big data manipulation, Biology direct, vol. 7, no. 1, p. 43, 2012.
[18] P. Mell and T. Grance, The nist definition of cloud computing, 2011.
[19] M. Simjanoska, A. M. Bogdanova, and Z. Popeska, Recognition of colorectal carcinogenic tissue with gene expression analysis using bayesian probability, in: ICT Innovations 2012. Springer, 2013, pp.305–314.
[20] M. Simjanoska, A. Madevska Bogdanova, and Z. Popeska, Bayesian posterior probability classification of colorectal cancer probed with affymetrix microarray technology, in: Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention on. IEEE, 2013, pp. 959–964.
[21] A. M. Bogdanova, M. Simjanoska, and Z. Popeska, Classification of colorectal carcinogenic tissue with different dna chip technologies, in: the 6th International Conference on Information Technology, ser. ICIT, 2013.
[22] M. Simjanoska and A. M. Bogdanova, Novel methodology for crc biomarkers detection with leave-one-out bayesian classification, in: ICT Innovations 2014. Springer, 2015, pp. 225–236.
[23] M. Simjanoska, A. M. Bogdanova, and Z. Popeska, Bayesian multiclass classification of gene expression colorectal cancer stages, in: ICT Innovations 2013. Springer, 2014, pp. 177–186.
[24] M. Simjanoska, A. M. Bogdanova, and S. Panov, Gene ontology analysis on behalf of improved classification of different colorectal cancer stages, Tech. Rep.
[25] ——, Gene ontology analysis of colorectal cancer biomarkers probed with affymetrix and illumina microarrays. in: Proceedings of the 5th International Joint Conference on Computational Intelligence, IJCCI, 2013, pp. 396–406.
[26] A. Hadoop, Apache hadoop, 2014. [Online]. Available: http://hadoop,apache,org.
[27] D. Hong, A. Rhie, S.-S. Park, J. Lee, Y. S. Ju, S. Kim, S.-B. Yu, T. Bleazard, H.-S. Park, H. Rhee et al., Fx: an rna-seq analysis tool on the cloud, Bioinformatics, vol. 28, no. 5, pp. 721–723, 2012.
[28] L. Jourdren, M. Bernard, M.-A. Dillies, and S. Le Crom, Eoulsan: a cloud computing-based framework facilitating high throughput sequencing analyses, Bioinformatics, vol. 28, no. 11, pp. 1542–1543, 2012.
[29] T. Gunarathne, T.-L. Wu, J. Y. Choi, S.-H. Bae, and J. Qiu, Cloud computing paradigms for pleasingly parallel biomedical applications, Concurrency and Computation: Practice and Experience, vol. 23, no. 17, pp. 2338–2354, 2011.
[30] R. C. Taylor, An overview of the hadoop/mapreduce/hbase framework and its current applications in bioinformatics, BMC bioinformatics, vol. 11, no. Suppl 12, p. S1, 2010.
[31] J. Gasior and F. Seredynski, Load balancing in cloud computing systems through formation of coalitions in a spatially generalized prisoner's dilemma game, in CLOUD COMPUTING 2012, The Third International Conference on Cloud Computing, GRIDs, and Virtualization, 2012, pp. 201–205.
[32] M. Simjanoska, M. Gusev, S. Ristov, and G. Velkoski, Scaling the performance and cost for elastic cloud web services, CIT. Journal of Computing and Information Technology, vol. 21, no. 2, pp. 85–95, 2013.
[33] M. Simjanoska, S. Ristov, and M. Gusev, Machine learning approach for performance based cloud pricing model, Proceedings of the 2013 International Conference on Applied Internet and Information Technologies, AIIT 2013, pp. 74–78.
[34] R. Rezaei, T. K. Chiew, S. P. Lee, and Z. Shams Aliee, A semantic interoperability framework for software as a service systems in cloud computing environments, Expert Systems with Applications, vol. 41, no. 13, pp. 5751–5770, 2014.
[35] T. Metsch, A. Edmonds, and V. Bayon, Using cloud standards for interoperability of cloud frameworks, SLA@ SOI, Tech. Rep, 2010.
[36] S. Dowell, A. Barreto, J. B. Michael, and M.-T. Shing, Cloud to cloud interoperability, in System of Systems Engineering (SoSE), 2011 6th International Conference on. IEEE, 2011, pp. 258–263.
[37] D. Petcu, C. Craciun, and M. Rak, Towards a cross platform cloud api, Components for Cloud Federation, Procs. CLOSER, pp. 166–169, 2011.
[38] K. Stravoskoufos, A. Preventis, S. Sotiriadis, and E. G. Petrakis, A survey on approaches for interoperability and portability of cloud computing services, 2014.
[39] Z. Zhang, C. Wu, and D. W. Cheung, A survey on cloud interoperability: taxonomies, standards, and practice, ACM SIGMETRICS Performance Evaluation Review, vol. 40, no. 4, pp. 13–22, 2013.