Algorithms for Process Mining – An Overview

Sandra Nikoloska
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
sandra.nikoloska@students.finki.ukim.mk

Biljana Sazdova
Faculty of computer science and engineering
Ss. Cyril and Methodius University in Skopje
Skopje, Macedonia
biljana.sazdova@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*—Process mining is a research area that provides an opportunity to utilize the gathered data about business processes that occur in organizations in order to provide improved business process models. That could be made by using process mining algorithms that analyze the event logs data from the business processes. Different research groups have proposed various algorithms for process mining by employing different approaches. This paper gives an introduction in the area of process mining, as well as an overview of the algorithms used within process mining, by describing how they work and how they are applied.**

Keywords— process mining, business process, business process intelligence.

I. Introduction

As business processes are automated with fast speed, the organizations have many data that are gathered, which could be utilized in order to improve the way how the organizations perform their work. The data from the business processes are in a form of event logs, so they present how the process executes in time for each instance (for example: a particular purchase order in the process for handling purchase orders). Since there is a huge amount and volume of data, it is not appropriate to analyze them manually. Therefore, there is a huge need for development of algorithms that could discover knowledge from these data that could be used to improve the business processes and the way how the work is done.

Process mining starts from the moment of data collection, their selection, analysis, and finally documentation. It is based on facts and is completely objective, and also it is made in automated manner thus we can save time and effort. In this way, there is no need for traditional methods for process discovery such as workshops, interviews and filtering of manual documentation, but by identifying the problems and finding solutions how they can be resolved based on the results obtained from the analyses that are made objectively and in real-time. Besides visualization of how the actual process flows, it also enables various analyses so that organizations can understand the attributes that influence the business processes and discover which activities they could improve. Process improvement, time reduction, cost reduction, increased transparency, improved customer experience, error reduction, better compliance and performance management are the main benefits of using process mining, which is a key component of business process management. It is necessary for the stakeholders to understand what their critical points are where problems arise and to rearrange the business processes for solving them [1].

There are several types of analyses that can be made in process mining, depending on the stage of the business process [2]. The most important are process discovery, compliance checking and model improvement. If these analyses are used together, they will help organizations maintain a high level of standardization, predictability, and continuous improvement. Organizations can use process mining to optimize a single process, individual department, or even the entire organization.

In this paper we give a short introduction into process mining, and we present the most important algorithms that could be used in order to discover business process models. The rest of this paper is organized in the following way. In section 2, we give an introduction into process mining algorithms and the types of problems that could be solved using them. Section 3 presents the Alpha Miner algorithm. Then, in section 4, the Heuristic Miner algorithm is described. The Genetic Miner algorithm is presented in section 5. Finally, section 6 concludes the paper.

II. Process Mining Algorithms

When we mention the word business, the things we initially think of are processes that give rise to problems, which we need to solve in the simplest way, and that in business is achieved with steps that will help us achieve the desired success. For that purpose, particular algorithm could be created, which shows how its steps (activities) can be defined as a series of rules or a sequence of well-defined instructions that are used to solve a particular problem. Within the business world, over time, various changes occur that can lead to slowing down the course of activities, the occurrence of unwanted inefficiencies, bottlenecks, increased costs, and so on. These changes are tracked and detected through process mining. Process mining can reveal better models and can help to design and redesign the processes within an organization. For example, if an organization wants to reduce its total customer interactions from 5% to 3%, with the help of process mining it will be revealed which parts can be optimized in order to achieve the desired goal [3].

Every organization wants to work efficiently and effectively, to produce more in less time, but to maintain the desired quality. To achieve this, it is necessary to provide well-defined business processes, which means discovering the dependencies between activities (for each activity to discover from which activity it is dependent on), in what order they can be performed, as well as the possibility of parallelization of certain activities that are part of the business process. For the success of each innovation in the organizations, it is important in each change to review the processes, and if necessary to redesign them with the aim of maximizing the effect of introduced change [4].

The main function of the algorithms used in process mining is to provide information that is part of the current operation that plays a key role in improving existing business processes, implementing a new operating model, audit, and

analysis [5]. The key to algorithms in this regard is to provide improvement.

The algorithms used in the process discovery help analysts to better understand the business processes and the problems that are part of them through the process models that result from the input data (event log) of the system [6].

In the next sections, we present some of the most important algorithms used in process mining, i.e. Alpha Miner, Heuristic Miner and Genetic Miner.

III. ALPHA MINER

Alpha Miner is the first algorithm to bridge the gap between event logs or monitored data and the discovery of a specific process model [7]. It is an algorithm that first appears in the process of process research (play in technique). This algorithm concisely determines the purpose of the research, adequately addresses the competition, and provides an overview (model) of the relationship between the activities that are an integral part of the process that is being examined. Each algorithm has a certain input that it processes and based on which in the end it produces an output or result. As input to the Alpha Miner algorithm, an event log is given, the algorithm analyzes the input, so based on the relations between the activities results in the construction of a graph or Petri network. The input data, i.e. the event log that is given to the Alpha Miner algorithm, consist of information about a specific activity or the whole process [8].

Here are some important facts about the Alpha Miner algorithm [8]:

- Uses a basic approach to process research.
- There are a number of limitations.
- Reveals key elements of the process.
- It is very important that the algorithm is used correctly using concrete examples.

L is a log of events resulting from the set of activities. T denotes trace where the activities are in a particular order. α(L) is the graph that represent the model of the business process, which is obtained with the following steps [9]:

(1) $T_L = \{t \in T \mid \exists_{\sigma \in L}\, t \in \sigma\}$

(2) $T_I = \{t \in T \mid \exists_{\sigma \in L}\, t = first(\sigma)\}$

(3) $T_O = \{t \in T \mid \exists_{\sigma \in L}\, t = last(\sigma)\}$

(4) $X_L = \{(A,B) \mid A \subseteq T_L \wedge A = \emptyset \wedge B \subseteq T_L \wedge B = \emptyset \wedge \forall_{a \in A} \forall_{b \in B}\, a \rightarrow_L b \wedge \forall_{a1,a2 \in A}\, a_1 \#_L a_2 \wedge \forall_{b1,b2 \in B}\, b_1 \#_L b_2\}$

(5) $Y_L = \{(A,B) \in X_L \mid \forall (A`,B`) \in X_L, A \subseteq A` \wedge B \subseteq B` \Rightarrow (A,B) = (A`,B`)\}$

(6) $P_L = \{p_{(A,B)} \mid (A,B) \in Y_L\} \cup \{i_L, o_L\}$

(7) $F_L = \{(a,p_{(A,B)}) \mid (A,B) \in Y_L \wedge a \in A\} \cup \{(p_{(A,B)},b) \mid (A,B) \in Y_L \wedge b \in B\} \cup \{(i_L,t) \mid t \in T_I\} \cup \{(t,o_L) \mid t \in T_O\}$

(8) $\alpha(L) = (P_L, T_L, F_L)$.

- ***Step 1:*** The first step is to determine all the events, i.e. to determine all the activities that are part of the event log (T_L).
- ***Step 2:*** Defining - discovering all the start activities (T_I), i.e. all the activities that appear first within the events.
- ***Step 3:*** Defining - discovering a set of end activities (T_O), i.e. all activities that appear at the end or are the last within the events.
- ***Step 4*** and ***Step 5:*** Define the basics of these algorithm. The main challenge in these two steps is to discover the set of activities and their connection, i.e. to discover their interrelationships. Within these steps, two sets A and B are created, set A represents input, while B represents output transitions. All activities that are part of the set A should be conditionally dependent on those that are part of set C. That is, (a, b) ∈ A x B: $a \rightarrow L^b$, the elements that are part of set A should not depend on each other, the same applies to those that are part of set C. In the 5th step, non-maximal pairs are removed, i.e. this step is also known as creating a relational matrix called Footprint matrix.
- ***Step 6:*** If the analysis of the relation matrix is done, it will be seen that for each element, i.e. pair there is a certain relation (→, #, ||). Basically, the basic goal of all these steps and globally the Alpha Miner algorithm is to determine the position and relation of the overall process.
- ***Steps 7 and 8:*** In these steps, the arcs are created - the relations between the activities, i.e. all the starting activities with the final transitions as an exit points.

The Alpha Miner algorithm can detect four types of relations [10], i.e. direct succession, causality, parallel (temporal independence), choice (no direct relation). These relations are described below.

- **Direct Succession**

Initial, basic and direct relation, on which the setting and execution of the other relations depend. This relation is denoted by the sign greater (>), which essentially means - "directly follows". **(a > b)** means that a is directly followed by b (the performance of activity b is always preceded by the performance of activity a). For example: from the event log [(a, b, c, d), (e, d, a, b), (a, e)], we can see that a and b are always in the same order, it is initially a, then b follows.

- **Causality**

This relation represents the temporal dependence between the activities. The relation is represented by the arrow sign (→). More specifically, if we take the activities a and b, **(a → b)** – in translation "b conditionally depends on a", i.e. this relation is valid if and only if (a> b) - "b directly follows a" and if (a >/ b) - "a never follows b". Basically, b depends on a.

- **Parallel – temporal independence**

Relation when certain activities are performed in parallel. This relation is denoted by two parallel vertical lines **(||)**, which essentially denote parallel execution. If there are two activities a and b and (a> b) - "b directly follows a" and (b>

a) - "a directly follows b", it can be concluded that these two activities are parallel, i.e. performed both activities in any order. For example, from the event log [(a, b, c, d), (e, d, b, a), (a, e)] we can see that a and b are activities that are directly monitored with each other, that means that they are temporally independent and can be performed in parallel [11].

- **Choice – No direct relation**

Independence, i.e. when a certain activity is independent. This relation is represented by the hash sign **(#)**. If we take the activities a and b, and neither b directly follows a neither a directly follows b, then the activities are completely independent. There is no direct relation between a and b (a # b), if neither a>b nor b>a. For example, from the event log [(a, d, c, b), (a, c, d, b), (a, e)] we can see that the activities a and b are independent [11].

IV. HEURISTIC MINER

The Heuristic Miner algorithm uses a similar representation as causal nets. It is important for this algorithm to note that event frequencies and sequences are taken into account when creating a process model. The basic idea of algorithm development is to minimize paths, i.e. those that are very rare to be completely omitted [12]. The Heuristic Miner algorithm is essentially an improvement of the Alpha Miner algorithm, the difference is that the frequencies of the activities are taken into account, and as a result it gives a model with complex dependencies in which the rare ones are left out. This algorithm is used in the research part of process discovery, it is very interesting and tempting to use due to its wide range of applications [4]. The main goal of the Heuristic Miner algorithm is to detect the causal net from the input event log [5].

The main improvements of Heuristic Miner over Alpha Miner are [3]:

- Frequencies are taken into account, which allows filtering of rare cases.
- Detection of short loops.
- Detection of skipping activities.

Heuristic Mining Algorithm uses dependency graph and causal matrix [13].

Dependency graph is a graph that shows the interdependencies of different activities and events. To build this model, first, you need a matrix in which the dependencies are displayed, and then it is necessary to determine the loops of length one and the loops of length two.

When analyzing event logs, it is difficult to determine whether certain activities or processes are parallel or sequential. A Heuristic Miner is an algorithm that builds a **causal matrix** in order to build an accurate and realistic process model.

There are two types of raw activities, AND and XOR. AND represents parallel activity, while XOR represents sequential activity.

When creating a graph dependency, thresholds could be set for the events and activities that would be modeled.

1. *Threshold* – is the minimum dependence between activities.
2. *Positive observation threshold* – is the minimum value of the frequencies between activities.
3. *Relative to the best threshold* – is the minimal difference between the value of the activity dependency and the maximum value (dependency value).
4. *Length-one threshold* – is the minimum dependency value of the same event.
5. *Length-two threshold* – is the minimum value of the loop dependency.

V. GENETIC MINER

Alpha Miner and Heuristic Miner create process models in a direct and determining way. On the other side, evolutionary approaches use iterative procedures to create and mimic the process of natural evolution. These types of approaches are not deterministic and depend on randomization in order to find new alternatives [14].

Same as the well-known genetic algorithms, Genetic Miner consists of four steps [14]: initialization, selection, reproduction, and termination. These steps are described below.

Initialization: Within the first step, the initial population is created. Process models are created randomly using the names of the activities that occur in the input event log. Process models that are part of the initial population do not have to initially correspond to the event log, i.e. the names of the activities are the same, but the behavior of the initial models will not be consistent with the input event log.

Selection: In this step, the "fitness" of each individual (model in this case) is calculated. The fitness function represents the quality of the model based on the event log. It is important to note that there are different ways to calculate the quality of a particular model. A simple criterion for determining the quality of a given model is to consider the proportion of traces in the event log that can be performed or repeated by the model. But this feature is not good because it is very likely that none of the models will be able to play any of the directions (paths) specified in the log. Also, if this type of criterion is used over universal models, then the fitness function will be too high. It is therefore necessary to use a more refined fitness function that rewards the partial correctness of the model and takes into account all four specified criteria. The best models of this generation, which are the models with highest fitness values, are passed on to the next phase, which is called elitism. In essence, only the best models go ahead and "survive", while the individuals (models) that do not have high fitness value are not retained, i.e. are considered as "dead" individuals.

Reproduction: As the name of the reproduction step itself states, it refers to the creation of new generation from the selected "parent" process models. As part of this step, two genetic operators are used, i.e.: cross-over and mutation. With cross-over two parent models are taken and used to create new ones which end up in a set of models - children. Children inherit from their parents, but they change to some extent as a result of mutations.

Termination: Reproduction and selection are repeated with the aspiration that each newly created generation will be

better than the previous one. The actual process of evolution ends at the moment when the created model is considered as a satisfactory solution. However, due to the event log, it will likely take a long time to determine the desired model, but it is also possible that the desired fitness value will not be found, so other criteria can be added and the reproduction for up to predefined number of generations can be made, so at the termination, the model with the best fitness value is returned.

VI. CONCLUSION

In this paper we gave a short introduction into process mining, as well as the benefits that it could provide to the organizations in order to rearrange how the work is performed by improving their business processes. We presented the most important algorithms that could be used to mine business process data that are gathered as event logs.

As described, the Alpha Miner algorithm discovers the temporal relations between the activities. Heuristic Miner extends this by considering the frequencies of the activities. This way, Alpha Miner and Heuristic Miner discover the process model in deterministic way. On the other side, the Genetic Miner uses randomization to randomly select initial population, as well as randomization to make reproduction (cross-over and mutation), thus it could give as an output different models in different trials.

ACKNOWLEDGMENT

This work was partially financed by the Faculty of computer science and engineering at the "Ss. Cyril and Methodius University in Skopje", Skopje, Macedonia.

REFERENCES

[1] M.Eisner, "What is Process Mining and Why Your Organization Needs It," 2020.

[2] "Why Use Process Mining," 2021, IBM Cloud Education.

[3] J.C.A.M. Buijs, Heuristics Miner, Technical Universiteit Eindhoven.

[4] A.P Kurnniati, G.P. Kusuma and G.A.A. Wisudiawan, "Implementing Heuristic Miner for Different Types of Event Logs," International Journal of Applied Engineering Research, vol. 11, no. 8, pp. 5523-5529, 2016.

[5] W.V.D Aalst, Process Mining: Data Science in Action, 2nd ed., Springer Berlin, Heidelberg, 2016, pp. 205.

[6] W.V.D Aalst, Process Mining: Data Science in Action, 2nd ed., Springer Berlin, Heidelberg, 2016, pp. 125.

[7] H. Simsek, "What are the Top 5 Process Mining Algorithms in 2022?," 2021.

[8] W.V.D. Aalst, "Alpha Algorithm: A Process Discovery Algorithm".

[9] W.V.D. Aalst, Process Mining: Data Science in Action, 2nd ed., Springer Berlin, Heidelberg, 2016, pp. 171-172.

[10] A. Rozinat "ProM Tips – Which Mining Algorithm Should You Use?," 2010.

[11] S. Conquest, "Alpha Algorithm (Process Discovery Method)," 2018.

[12] W.V.D. Aalst, Process Mining: Data Science in Action, 2nd ed., Springer Berlin, Heidelberg, 2016, pp. 201.

[13] A.J.M.M. Weijters, W.M.P van der Aalst, and A.K. Alves de Medeiros, Process Mining with The HeuristicsMiner Algorithm, Technische Universiteit Eindhoven, 2013.

[14] W.V.D. Aalst, Process Mining: Data Science in Action, 2nd ed., Springer Berlin, Heidelberg, 2016, pp. 207-210.