Process Mining A Comparative Study

Conference Paper · December 2014

DOI: 10.17148/ijarcce

CITATIONS
26

READS
2,527

2 authors, including:

Esmita Gupta
Vidyalankar Institute of Technology
4 PUBLICATIONS **32** CITATIONS

SEE PROFILE

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 11, November 2014

Process Mining A Comparative Study

Asst. Prof. Esmita .P. Gupta

M.E. Student, Department of Information Technology, VIT, Mumbai, India

Abstract: The systems that support today's globally distributed and agile businesses are steadily growing in size and generating numerous events. Business Intelligence aims to support and improve decision making processes by providing methods and tools for analyzing the data. Process mining builds the bridge between Data Mining as a Business Intelligence approach and Business Process Management. Its primary objective is the discovery of process models based on available event log data.
Many process mining algorithms have been proposed recently, there does not exist a widely-accepted benchmark to evaluate and compare these process mining algorithms. As a result, it can be difficult to choose a suitable process mining algorithm for a given enterprise or application domain.
This paper proposes a solution to evaluate and compare these process mining algorithms efficiently, so that businesses can efficiently select the process mining algorithms that are most suitable for a given model set.

Keywords: Process mining, Heuristic miner, Genetic miner, Fuzzy miner

I. INTRODUCTION

Most organizations use information systems to support the execution of their business processes. Examples of information systems supporting operational processes are Workflow Management Systems (WMS), Customer Relationship Management (CRM) systems and Enterprise Resource Planning (ERP) systems and so on. These information systems may contain an explicit model of the processes may support the tasks involved in the process without necessarily defining an explicit process model [9].
Process mining is a relatively young research discipline that sits between computational intelligence and data mining on the one hand, and process modeling and analysis on the other hand. The idea of process mining is to discover, monitor and improve real processes by extracting knowledge from event logs readily available in today's (information) systems [3].

The main benefit of process mining techniques is that information is objectively compiled. In other words, process mining techniques are helpful because they gather information about what is actually happening according to an event log of an organization, and not what people think that is happening in this organization.

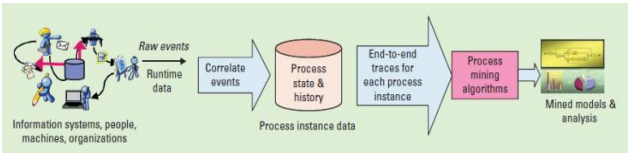

Fig 1: The process-mining pipeline, from executing processes to mining and analytics[2]

Figure 1 shows an overview of the pipeline from processes executing in the real world to process-mining algorithms. The process starts with the collection of all the events and correlating raw events from executing processes to isolate end-to-end instances of the same process. This generates an execution trace corresponding to a process instance. An execution trace is a recorded collection of events in a sequential manner, where each event refers to a task and is related to a particular case (a process instance).

Here, we provide a short introduction to process mining, including a comparison of three established process-mining algorithms. Different process mining algorithms have different properties (e.g., some perform better on models with invisible tasks, while others do not), being able to select the most appropriate process mining algorithm (i.e. the one that produces mined models that are semantically similar to the original models and structurally equal to or better than the original models) for the models from a given enterprise is a big advantage [5].

The organization of the rest of this paper is as follows. Section 2 discusses why process mining. Section 3 discusses about the working of heuristic, genetic and fuzzy miner algorithm. The comparative study is given in section 4.

II. WHY PROCESS MINING

The emergence of semi-structured processes, combined with improvements in computing power and the speed of data transmission, have fueled the need for mining algorithms to address new challenges.
In addition to discovering process models from logs and examining the level of conformance of an actual process to its modeled counterpart, process mining should find, merge, and clean event data; handle changes in a process that occur while its being mined; and provide operational support to process users in an online manner.[1]

1. The first challenge is that numerous uncorrelated events (with possible noise) are gathered from disparate heterogeneous and distributed systems.
2. The second challenge is the Dramatic variation in execution behavior.
3. The third challenge is parallel and repeated task execution.

Thus this topic includes a method for systematic comparison, and a body of experimental data describing the behavior of three algorithms i.e. Heuristic Miner, Fuzzy miner and Genetic Miner with typical process structures and how it can handle the above challenges.

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 11, November 2014

The primary aim of process mining is to investigate a scalable solution that can evaluate, compare and rank these process mining algorithms efficiently. In particular, it attempts to investigate how we can choose an effective process mining algorithm for an enterprise without evaluating different process mining algorithms.

III. PROCESS MINING ALGORITHM

There are many different approaches to process mining. Local methods look at local relations between activities in logs(Heuristics Miner), while global approaches build and refine a model based on the whole log (Genetic Mining, Fuzzy Miner) Different algorithms have their own specialism. Heuristics Miner uses frequencies and parameterization to handle noise; while Genetic Process Mining can mine complex and noisy logs, but is resource intensive. More recent approaches focus on managing complex real world models or noisy logs using clustering and abstraction, e.g. at the trace or activity level.
In this paper we are going to objectively differentiate the most used algorithms with examples.

1. Heuristic miner
2. Genetic miner
3. Fuzzy miner

A. Heuristic Miner

Heuristics Miner is a practical applicable mining algorithm that can deal with noise, and can be used to express the main behavior that is not all details and exceptions, registered in an event log. This technique extends alpha algorithm by considering the frequency of traces in the log. The Heuristics Miner Plug-in mines the control flow perspective of a process model. To do so, it only considers the order of the events within a case. For instance for the log in the log file only the fields case id, time stamp and activity are considered during the mining. The timestamp of an activity is used to calculate these orderings [9].

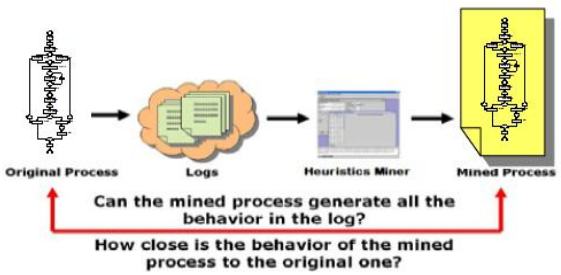

Fig 2: Heuristic Process Mining [9]

1) Steps of Heuristic Miner algorithm:

1. Read a log.
2. Get the set of tasks.
3. Infer the ordering relations based on their frequencies.
4. Build the net based on inferred relations.
5. Output the net [12].

As we know, Process mining techniques are able to extract knowledge from event logs. To find a process model on the basis of an event log, the log should be analyzed for causal dependencies, e.g., if an activity is always followed by another activity it is likely that there is a dependency relation between both activities. To analyze these relations we introduce the following notations.
Let T be a set of activities. S ϵ T is an event trace, i.e., an arbitrary sequence of activity identifiers. W Ɛ T is an event log, i.e., a multiset that is bag of event traces. Note that since W is a multiset, every event trace can appear more than once in a log [9].
Let's analyze these relations by introducing the following notations. Let W be an event log over T, i.e., W ϵ T. Let a, b ϵ T:

1. a ⇒ W b if and only if there is a trace S = t1,t2,t3 :: : tn and i ϵ {1; : : : ; n -1} such that S ϵ W and ti= a and ti+1 = b.
2. a →W b if and only if a ≯W b and b >W a
3. a ≠ W b if and only if a ≯W b and b ≯ W a.
4. a ‖ W b if and only if a >W b and b >W a.
5. a>>W b if and only if there is a trace S = t1,t2,t3….. tn and i Ɛ {1; : : : ; n - 2} such that S ϵ W and ti= a and ti+1 = b and ti + 2 = a.
6. a>>>W b if and only if there is a trace S = t1, t2,t3….. tn and i < j and i, j ϵ {1……n} such that S ϵ W and ti = a and tj = b.

After inferring the relation according to the frequencies we start with the construction of a so called *dependency graph*. A frequency based metric is used to indicate how certain we are that there is truly a dependency relation between two events 'a' and 'b' (notation a⇒Wb).The calculated ⇒W values between the events of an event log areused in a heuristic search for the correct dependency relations.

1. Let W be an event log over T, and a, b ϵ T. Then |a >W b|is the number of times a >W b occurs in W, and,

$$a \Rightarrow w\ a = \left(\frac{|a>wb|-|b>wa|}{|a>wb|+|b>wa|+1}\right) \qquad (1)$$

Where a⟹Wb is always between 1 and -1.

For **short loops**, the dependency is measured as follows, Let W be an event log over T, and a, b ϵ T Then |a >W a| is the number of times a >W a occurs in W, and |a >>W a| is the number of times a >>W b occurs in W.

$$a \Rightarrow w\ a = \left(\frac{|a>wa|}{|a>wa|+1}\right) \qquad (2)$$

$$a \Rightarrow_{2W} b = \left(\frac{|a\gg wb|+|b\gg wa|}{|a\gg wb|+|b\gg wa|+1}\right) \qquad (3)$$

2. **AND/XOR-split/join and non-observable tasks**:
Mining of non-observable activities is difficult, because they are not present in the event log. To avoid the explicit modeling of invisible activities, in the Heuristics Miner we do not use Petri nets for the representation of process models, but so called Causal Matrix.
Hence, Heuristics Miner is a most practical applicable mining algorithm that can deal with noise, and can be used to express the main behavior that is not all details and exceptions, registered in event logs [14].

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 11, November 2014

B. Genetic Miner

Genetic mining algorithms use an evolutionary approach that mimics the process of natural evolution. They are not deterministic.

A genetic search is an example of a global search strategy because the quality or fitness of a candidate model is calculated by comparing the process model with all traces in the event log the search process takes place at a global level. For a local strategy there is no guarantee that the outcome of the locally optimal steps (at the level of binary event relations) will result in a globally optimal process model. Hence, the performance of such local mining techniques can be seriously hampered when the necessary information is not locally available because one erroneous example can completely mess up the derivation of a right model. Therefore, we started to use genetic algorithms.

Genetic algorithm is used to discover a Petri net given a set of event traces. Genetic algorithms are adaptive search methods that try to mimic the process of evolution [9].

These algorithms start with an initial population of individuals (in this case process models). Populations evolve by selecting the fittest individuals and generating new individuals using genetic operators such as crossover (combining parts of two of more individuals) and mutation (random modification of an individual) [8].

The goal of using genetic algorithms is to tackle problems such as duplicate activities, hidden activities, non-free-choice constructs, noise, and incompleteness, i.e., overcome the problems of some of the traditional approaches.

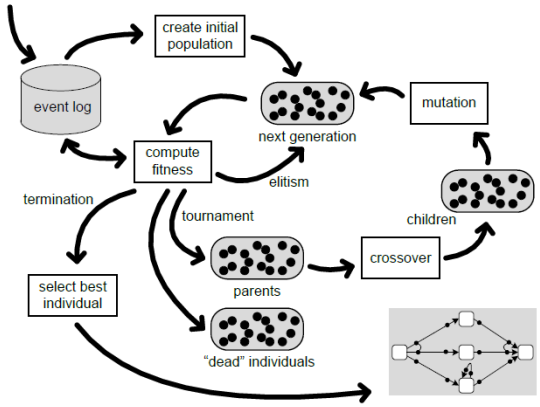

Fig 3: Genetic Miner algorithm [12]

1) Genetic Miner algorithm:

The most important building blocks of our genetic approach: (i) the initialization process, (ii) the fitness measurement, and (iii) the genetic operators.

Main steps of our genetic algorithm are:

1. Read the event log.
2. Calculate dependency relations among activities. Build the initial population.
3. Calculate individuals' fitness.
4. Stop and return the fittest individuals?
5. Create next population by using the genetic operators [12].

Thus Genetic mining algorithm requires a lot of computing power which can be distributed easily, can deal with noise, infrequent behavior, duplicate tasks and invisible tasks. Allows for incremental improvement and combinations with other approaches.

c. Fuzzy Miner

Real-life processes turn out to be less structured model than people tend to believe. Unfortunately, traditional process mining approaches have problems dealing with unstructured processes. The discovered models are often "spaghetti-like", showing all details without distinguishing what is important and what is not. Fuzzy miner algorithm is configurable and it allows for different faithfully simplified views of a particular process. To do this, the concept of a roadmap is used as a metaphor [10].

Fuzzy Miner is suitable for mining less structured processes which exhibit a large amount of unstructured and conflicting behavior i.e. spaghetti-like models into more concise models. It applies a variety of techniques, such as removing unimportant edges, clustering highly correlated nodes in to a single node, and removing isolated node clusters. [2]

An Adaptive Approach for Process Simplification

Process mining techniques which are suitable for less-structured environments need to be able to provide a high-level view on the process, abstracting from undesired details.

- Activities in a process can be related to locations in a topology (e.g. towns or road crossings) and
- Precedence relations to traffic connections between them (e.g., railways or motorways).

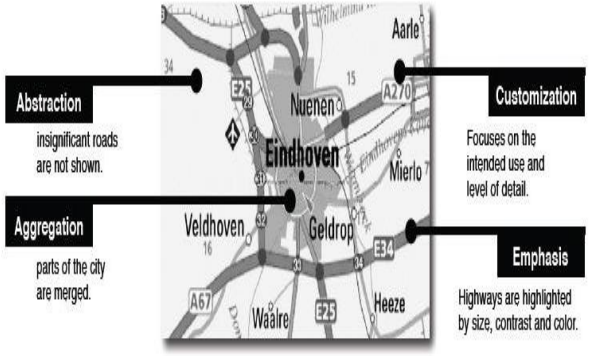

Fig 4. Example of Road Map [13]

1. Fuzzy miner algorithm:

Input

A set of N transactions, each with n attribute, fuzzy linguistic terms for quantitative attributes, The user-specified minimum fuzzy support, The user-specified minimum fuzzy confidence, A domain Ontology.

Output

Phase I: Fuse similar behaving attributes;
Phase II: Generate Meta rules;
Phase III: Generate frequent fuzzy itemsets;
Phase IV: Make fuzzy association rules [15].

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 11, November 2014

Fuzzy Miner has been enhanced to utilize the availability of sub-logs obtained from the Pattern Abstractions plugin for the chosen abstractions. Fuzzy models are discovered for each of the sub-logs and are displayed upon zooming in on its corresponding abstract activity. Abstract activities are differentiated from other activities by means of a distinct color.

Thus we can conclude saying that we can use this miner algorithm approach to create maps that

- depict desired traits,
- eliminate irrelevant details,
- reduce complexity, and
- Improve comprehensibility [7].

IV. COMPARATIVE STUDY OF MINING ALGORITHM

Miner algo / characteristic	**Heuristic Miner**	**Fuzzy Miner**	**Genetic miner**
Description	Provides a view of scientific workflows by considering long distance dependency	Provides a zoomable view of scientific workflow by controlling significance cutoff to show task at different importance level	Provides a view of frequency for both tasks and succession between both tasks, and discovers all common control-flow structures.
Strategy	Work based on local strategy technique to build a model	Work based on both local and global strategy technique to build a model	Work based on global strategy technique to build a model
Output	Heuristic Net	Fuzzy model	Petri net graph
When to use it	When you have real-life data with not too many different events	When you have complex and unstructured log data, or when you want to simplify the model in an interactive manner	When you need to generate a random population of process models and to find a satisfactory solution
Challenging Problems	Can mine logs which are less sensitive to noise, local and nonfree choice constructs.	Can mine logs with noise, but cannot be converted to petri net.	Can mine logs with noise, handling of duplicate task names, local and nonlocal nonfree choice constructs and invisible task.
Behavior	Heuristic mining algorithms take frequencies into account	Fuzzy miner uses dependency graph representation	Genetic mining algorithms mimic natural evolution
Format	HM can mine Unstructured process	FM can mine less structured process	GM can mine both structured and Unstructured process
Log Data	HM can handle incomplete logs to a certain extent	FM can handle incomplete logs	GM can easily handle incomplete logs.
Result	HM gives long distance dependency successfully, but gives too much dependency for some task	FM successfully gives the changed part and unchanged part, giving the most dependency correctly.	GM gets a good view of structures and frequencies, yet giving some wrong dependency which does not exist in the original scientific workflow

V. CONCLUSION

The significance of process mining increases with the growing integration of information systems.

Many process mining techniques have been proposed. It is difficult to know which algorithms are better.

Thus here we have carried out a comparative study to identify which algorithm to be used and when. After the study we can say that

* Heuristic miner can be used when we have real-life data with not too many different events or when you need a Petri net model for further analysis

* Genetic Miner can be used when we need to mine logs with noise, handling of duplicate task names, local and nonlocal non free choice constructs and invisible task.

* Fuzzy miner can be used when we have seen various algorithms (i) depict desired traits, (ii) eliminate irrelevant details, (iii) reduce complexity, and (iv) improve comprehensibility.

Process mining is still a young research discipline. Although a model mined by a process-mining algorithm can present a picture of a process execution at a particular point in time, as the process changes and evolves, the mined model might no longer maintain fidelity to the process. This leads to the challenge of handling changes in the process (also known as concept drift) as the process is being analyzed. The future scope of our topic will be to have runtime changes in the process model with the help of these algorithms with monitoring, verification of requirements, and compliance enforcement.

REFERENCES

[1] IEEE Taskforce "Process mining manifesto", 2012
[2] Geetika T. Lakshmanan and Rania Khalaf, "Leveraging Process mining techniques", IBM T.J. Watson Research Center , IT Pro September/October 2013
[3] StB Prof. Dr. Nick Gehrke, Michael Werner, Dipl.-Wirt.-Inf. "Process Mining", WISU - die Zeitschriftfür den Wirtschaftsstudenten 7/13
[4] Philip Weber, Behzad Bordbar, Peter Tino, Basim Majeed "A Framework for comparing Process Mining Algorithm"

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

[5] Jianmin Wang, Raymond K. Wong, Jianwei Ding, Qinlong Guo and Lijie Wen "Efficient selection of mining algorithm", IEEE Transactions on Services Computing 01/2013
[6] RengZeng, Xudong He, Jiafei Li, Zheng Liu, W.M.P. van der Aalst "A Method to Build and Analyze Scientific Workflows from Provenance through Process Mining", Proceedings of the 3rd USENIX Workshop on the Theory and Practice of Provenance (TaPP'11); 01/2011
[7] R.P. Jagadeesh Chandra Bose, Eric H.M.W. Verbeek1 and Wil M.P. van der Aalst1 "Discovering Hierarchical Process Models Using ProM", Department of Mathematics and Computer Science, University of Technology, Eindhoven, The Netherlands, Conference: Proceedings of the CAiSE Forum 2011
[8] W.M.P. van der Aalst, A.K. Alves de Medeiros, and A.J.M.M. Weijters "Genetic Process Mining", Department of Technology Management, Eindhoven University of Technology, 26th International Conference, ICATPN 2005
[9] Saravanan .M.S, Rama Sree .R.J "A Role of Heuristics Miner Algorithm in the Business Process System", Comp. Tech. Appl., Vol 2 (2), 340-344
[10] W.M.P. van der Aalst*, A.J.M.M. Weijters "Process mining: a research agenda", Department of Technology Management, Eindhoven University of Technology,2004.
[11] Fabrizio Maria Maggi "Process Mining-Control flow process discovery", Springer 2011
[12] Ana Karla Alves de Medeiros," Process Mining-Control flow mining algorithm", Eindhoven University of Technology
[13] Christian W. Gunther and Wil M.P. van der Aalst "Fuzzy Mining – Adaptive Process Simplification Based on Multi-Perspective Metrics", Eindhoven University of Technology.
[14] A.J.M.M. Weijters, W.M.P. van der Aalst, and A.K. Alves de Medeiros "Process Mining with the Heuristics Miner Algorithm"
[15] Zahra Farzanyar, Mohammadreza Kangavari, "Efficient Mining of Fuzzy association Rules from Preprocessed Dataset", Computing and Informatics, Vol. 31, 2012, 331–347
[16] Dr. Anne Rozinat and Dr. Christian W. Günther "The Added Value of Process Mining", 2014.
[17] "Introduction to Process Mining, turning big data in real value.mp4", www.processmining.org, 04 Aug 2014.
[18] "Fuzzy miner", www.fluxicon.com, 13 Sept 2014.
[19] "Which mining algorithm should you use", www.processmining.org, 04 Aug 2014.
[20] "How to get started with process mining", www.processmining.org, 04 Aug 2014.