

Process Mining – New era in process management - A Survey

By

Divya Kundra, Prerna Juneja

Assistant Professor, Deen Dayal Upadhyaya College, Delhi, India

Software Engineer, EMC, Bangalore, India

divyakundra21@gmail.com, prerna1399@iiitd.ac.in

ABSTRACT

Process Mining is extraction of process models from event logs recorded by information systems. It acts as a bridge between data mining and business process. It is possible to use process mining to monitor deviations (e.g., comparing the observed events with predefined models or business rules in the context of SOX). This technology has become available only recently, but it can be applied to any type of operational processes (organizations and systems). Example applications include: analyzing treatment processes in hospitals, improving customer service processes in a multinational, understanding the browsing behavior of customers using a booking site, analyzing failures of a baggage handling system, and improving the user interface of an X-ray machine. The paper does a survey of important process mining aspects- its perspective, types, tools available for it, advantages and the fundamental algorithm used in Process Mining- Alpha Miner Algorithm.

Keywords

Event logs, Alpha Miner, Process Aware Information System, Disco.

1. INTRODUCTION

Process Aware Information System (PAIS) are software systems managing and executing operational processes that involves people, applications and information sources on the basis of process models within an organization [1]. Process mining is extraction of valuable insights from event logs produced by PAIS. Examples of PAIS systems includes Workflow Management Systems (WMS) and Business Process Management Systems (BPMS). An event log stores the detailed information about each development represented as an event which is executed in the process. An event is stored as the combination of 4 fields: activity (well defined step in a process), Case Id, (identifier, for which activity is recorded), actor (person starting and performing the activity) and timestamp (the beginning time of the event) [2]. A trace/case/process-instance is defined as the sequential representation of all the activities belonging to the same Case Id in order of their occurrence with respect to time. A process model is a visual representation of the process which is recorded by event log [2]. The event logs produced by PAIS systems are used to produce process models as shown in Figure 1.

Process models gives clear-cut business insights by presenting a directed graph in terms of nodes and edges. Nodes represent the activities that are performed in the process while edges connect the activities in order of their occurrence with respect to time. If an activity A has an earlier timestamp than activity B, then there will be an edge from node representing activity A to node representing activity B. There are several notations to represent the process models such as petri nets [3], causal nets [4] etc. A petri net [4] is also known as place/transition net. While a transition represent the activity executed in the process, place denotes the pre or post conditions that must be followed to reach from one transition to another. Petri net is represented by 3 tuple value (S, T, W) where S and T are finite set of places and transitions respectively. W is a multi set of arcs. One arc can connect one place and transition only. As seen from Figure 2, places are represented with circles, transitions are activities represented inside rectangles and arcs are denoted by directed arrows that show connection between places and transitions or vice-versa.

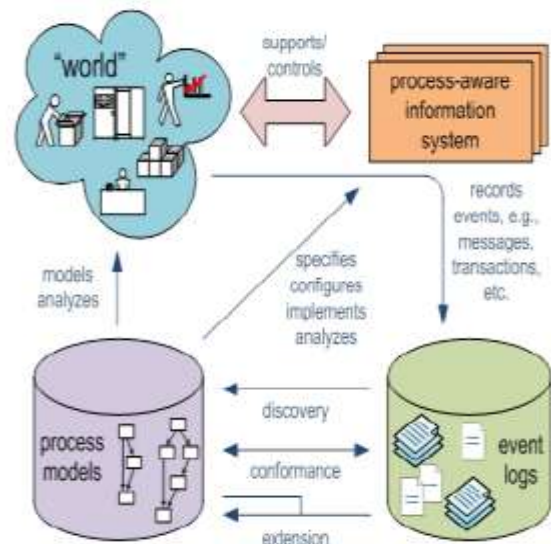


Figure 1 : Types of Process Mining

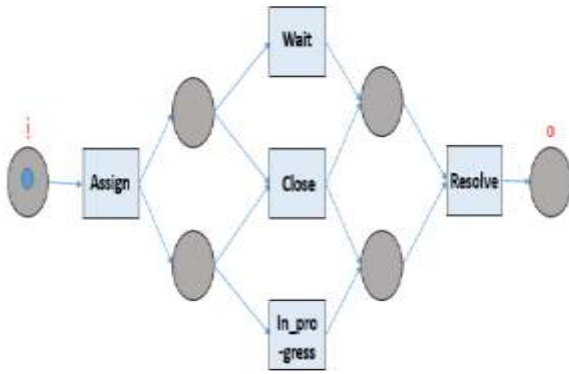


Figure 2 : Petri Net consisting of places and transitions.

1.1 Process Mining Perspectives

Following are the three perspectives of process mining [5]:

1. **Process Perspective [5]:** This perspective aims at producing process models from the event log focusing on order of the occurrence of events. It is also known as Control flow perspective. The aim of this perspective is to find good characterisation of all the possible paths that exists between the nodes representing activities.
2. **Organisational Perspective [5]:** This perspective focuses on determining the structure of the organisation from people involved, their roles and relationships. It focuses on the actor field of the event log.
3. **Case Perspective [5]:** This perspective focuses on properties of cases. It aims at analysing properties of cases like the count of involved activities, represented Case Id for particular set of activities etc.

1.2 Process Mining Types

As shown in Figure 1 process mining can be of the following types [1]:

1. **Discovery [1]:** It takes an event log as an input to produce process model for the recovery of hidden and valuable information. It does not use any a-priori information
2. **Conformance [1]:** It checks if the process model and event log have conformance with each other. It helps in monitoring all the deviations that occur in organisation's process.
3. **Enhancement [1]:** It aims at improvement of the existing process. Generated process models may indicate the need for improving the current standard of the workflow. Whereas conformance checking measures the alignment between model and reality, this third type of process mining aims at changing or extending the a-priori model. An example is the extension of a process model with performance information, e.g., showing bottlenecks. Process mining techniques can be used in an offline, but also online setting. The latter is known as operational support. An example is the detection of non-conformance at the moment the deviation actually takes place.

Another example is time prediction for running cases, i.e., given a partially executed case the remaining processing time is estimated based on historic information of similar cases.

1.3 Advantages of Process Mining

Some of the advantages of process mining are understanding and identifying the root causes behind problems, locating anomalies, finding bottlenecks by detecting the regions (nodes or edges) where the largest delays occur, achieving compliance by quantifying the amount by which the process models deviate from real-life, validating process changes and understanding the entire process.

1.4 Tools of Process Mining

There are many software tools available to mine the event logs e.g. Disco, Prom etc. ProM (which is short for Process Mining framework) is an Open Source framework for process mining algorithms. ProM provides a platform to users and developers of the process mining algorithms that is easy to use and easy to extend. Disco automatically creates insightful process models providing a lot of advanced features to make things simpler to understand by a process analyst. The files that can be imported must be in CSV, XLS, MXML or XES format. Figure 3 shows an event log being imported in Disco to form a process model. Column 1 indicates the case id of each event, Column 2 the actor, Column 3 the timestamp and Column 4 the activity. Each column of the event log has to be selected and configured either as the Case Id, Activity, Actor (Resource), or the Timestamp. Atleast Case Id and Activity column has to be mentioned in the log without which the log would not be acceptable for the import. The pattern of timestamp entered has to be selected from the timestamps already available or it can be specified in Java's Simple Date format. In Disco, a zoom slider is provided that gives full control to the user to dig deeper into the model. Due to complex and confusing nature of real life event logs, Disco shows only the most important flows that is the most frequently occurring variants. Sliders for adjusting the number of activities and number of paths that are to be presented in the model are available. The process model has a starting (represented by a triangle symbol) and an end node (represented by a stop symbol) with all the activities, whose names are mentioned in the node itself, lying in between the two. Dashed arrows point to activities that occur at the very beginning or very end of the processes. Absolute frequency of each transition is written over them, nodes with more darker color and thicker edges signify a higher frequency count. Figure 4 shows a process model generated from an event log using Disco.

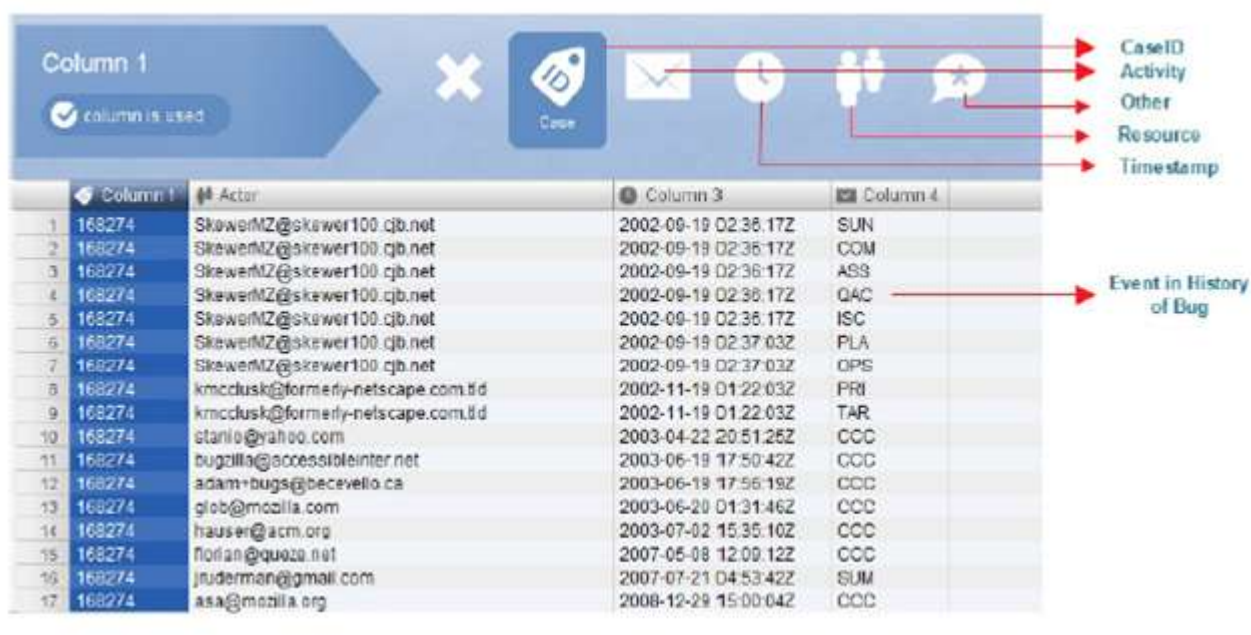


Figure 3: Snapshot of Disco Tool

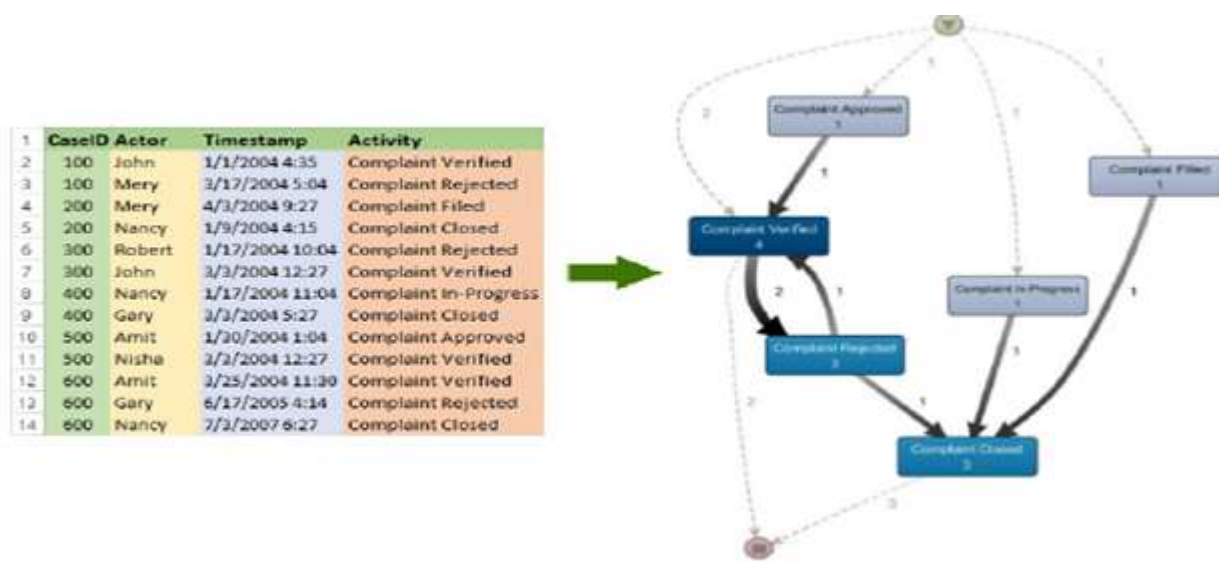


Figure 4 : Generating Process model from Event Log

1.5 Alpha Miner Algorithm

Alpha Miner is one of the most popular process mining algorithm that was proposed by Wil van der Aalst, Ton Weijters, and Laura Maruster in 2004. It is a fundamental process discovery algorithm that extracts a process model from an event log

consisting of traces and represents it as a petri net [6]. It is based on analyzing immediate successor relation between activities present in the event log. Table 1 shows an event log which is converted to a sequential event log L as shown in Table 2 in which the activities belonging to the same Case Id appear sequentially in increasing order of timestamp

Table 1 Event Log L

Case Id	Activity
1	Assign
1	Wait
1	In_progress
1	Resolve
2	Assign
2	Close
2	Resolve
3	Assign
3	In_progress
3	Wait
3	Resolve

Table 2 Sequential Event Log

Activity
Assign, Wait, In_progress, Resolve
Assign, Close, Resolve
Assign, In_progress, Wait, Resolve

The algorithm scans through a sequential event log like one shown in Table 1 and establishes the following relations [7] between all the activities:

1. **Direct Succession** ($a > b$) if activity a directly precedes activity b in some trace i.e. activity a having earlier timestamp value is immediately followed by activity b which is performed just after a .
2. **Causal** ($a \rightarrow b$) if $a > b$ and $b \not> a$.
3. **Parallel** ($a \parallel b$) if $a > b$ and $b > a$.
4. **Unrelated** ($a \# b$) if $a \not> b$ and $b \not> a$.

For the sequential event log L shown in Table 2 the pair of activities having direct succession for event log L is displayed in Table 3. Table 4 reports the Causal relation for event log L, Table 5 represents the parallel relation for event log L which is also computed with the help of direct succession. Table 6 reports the unrelated relation for event log L computed with the help of direct succession.

Table 3 Direct Succession Relation

Activity a	Activity b
Assign	Wait
Wait	In_progress
In_progress	Resolve
Assign	Close
Close	Resolve
Assign	In_progress
In_progress	Wait
Wait	Resolve

Table 4: Causal Relation

Activity a	Activity b
Assign	Wait

Assign	In_progress
Assign	Close
Wait	Resolve
In_progress	Resolve
Close	Resolve

Table 5 : Parallel Relation

Activity a	Activity b
Wait	In_progress
In_progress	Wait

Table 6: Unrelated Relation

Activity a	Activity b
Assign	Resolve
Wait	Close
In_progress	Close
Resolve	Assign
Close	Wait
Close	In_progress
Assign	Assign
Wait	Wait
In_progress	In_progress
Resolve	Resolve
Close	Close

Let the event log L shown in Table 2 be over T activities or transitions. The detailed steps involved in the Alpha Miner algorithm are as follows:

1. Determine the set of unique activities present in the event log. In event log L, it is {Assign, Wait, In progress, Resolve and Close}.
2. From the set of all activities T, determine the set of activities which do not have immediate predecessor anywhere in any of the trace in the log. In event log L, it is {Assign}.
3. From the set of all activities T, determine the set of activities which do not have immediate successor anywhere in any of the trace in the log. In event log L it is {Resolve}
4. Scan through the traces present in the sequential event log and determine the above mentioned relations ($>$, $\rightarrow$, $\parallel$, $\#$) between all activities and represent them in the form of a matrix called footprint. Table 7 shows footprint matrix for L

Table 7: Footprint Matrix

	Assign	Wait	In_prog ress	Resolve	Clos e
Assign	#	$\rightarrow$	$\rightarrow$	#	$\rightarrow$
Wait	$\leftarrow$	#	$\parallel$	$\rightarrow$	#
In_progre ss	$\leftarrow$	$\parallel$	#	$\rightarrow$	#
Resolve	#	$\leftarrow$	$\leftarrow$	#	$\leftarrow$
Close	$\leftarrow$	#	#	$\rightarrow$	#

5. Using the footprint matrix we generate XL that consists all possible pairs of sets (A,B) such that activities within set A and within set B are unrelated to rest of the activities of their set and each activity in set A is in causal relation with every activity of set B. Table 8 reports XL for L.

Table 8: Set Pair (A,B)

Set A	Set B
Assign	Wait
Assign	In_progress
Assign	Close
In_progress	Resolve
Assign	{Wait, Close}
Assign	{In_progress, Close}
Close	Resolve
{Wait, Close}	Resolve
{In_progress, Close}	Resolve
Wait	Resolve

6. In XL, if for a set pair (A,B), all activities in A are a subset of activities in set A' and all activities in set B are a subset of activities in set B' and (A', B') set pair is present in XL, then set pair (A,B) in XL is considered to be same as set (A',B'). Table 9 reports YL for L.

Table 9: Maximal Set Pair (A,B)

Set A	Set B
Assign	{Wait, Close}
Assign	{In_progress, Close}
{Wait, Close}	Resolve
{In_progress, Close}	Resolve

7. In PL, a place (discussed in Section 1) is generated for each distinct pair of set (A,B). Along with it, a input place iL and output place are oL generated.

8. For each set pair (A,B) in YL, arcs are connected from every activity present in set A to a place generated for the set pair (A,B) and arcs are also connected from the place to every activity present in set B. Activities in TI are connected to the input place iL and activities in TO are connected to the output place oL.

9. The generated petri net of Alpha Miner algorithm is represented by PL, TL and FL as shown in Figure 2

5 CONCLUSION

In this paper, a review of essential aspects of Process Mining - its perspective, types, tools available for it, advantages and its fundamental algorithm- Alpha Miner Algorithm is done. The study presented in the paper can be used by any novice learner in the field of process mining to enhance the crucial understanding of process mining.

3. REFERENCES

- [1] Wil M. Aalst. Transactions on Petri Nets and Other Models of Concurrency II. chapter Process-Aware Information Systems: Lessons to Be Learned from Process Mining, pages 1{26. Springer-Verlag, Berlin, Heidelberg, 2009.
- [2] Wil van der Aalst. Process Mining: Overview and Opportunities. ACM Trans. Manage. Inf. Syst., 3(2):7:1{7:17, July Heidelberg, 2009.
- [3] James L. Peterson. Petri Nets. ACM Comput. Surv., 9(3):223-252, September 1977.
- [4] Wil Van Der Aalst, Arya Adriansyah, and Boudewijn Van Dongen. Causal Nets: A Modeling Language Tailored Towards Process Discovery. In Proceedings of the 22Nd International Conference on Concurrency Theory, CONCUR'11, pages 28{42, Berlin, Heidelberg, 2011 Springer-Verlag
- [5] W. M. P. van der Aalst, H. A. Reijers, A. J. M. M. Weijters, B. F. van Dongen, A. K. Alves de Medeiros, M. Song, and H. M. W. Verbeek.
- [6] Business Process Mining: An Industrial Application. Inf. Syst., 32(5):713:732, July 2007.
- [7] Wil Van der Aalst, Ton Weijters, and Laura Maruster. Workflow mining: Discovering process models from event logs. Knowledge and Data Engineering, IEEE Transactions on, 2004.