International Journal of Applied Engineering Research ISSN 0973-4562 Volume 11, Number 8 (2016) pp 5523-5529
© Research India Publications. http://www.ripublication.com

Implementing Heuristic Miner for Different Types of Event Logs

Angelina Prima Kurniati[1], GunturPrabawa Kusuma[2], GedeAgungAry Wisudiawan[3]
[1,3]School of Computing, Telkom University, Indonesia.
[2]School of Applied Science, Telkom University, Indonesia.
[1,2,3]Jalan Telekomunikasi, TerusanBuahBatu, Bandung, Indonesia.

Abstract

Heuristic Miner is a stable process mining algorithm that is proven to be well-implemented in many cases. Process mining is a relatively new study which tries to extract information from event logs which is automatically recorded by the information system. It is said that process mining analyzes the "real process", because event logs record everything that are actually done by the users. This paper reports our analysis after implementing heuristic miner for different types of event logs. Heuristic miner is implemented for process mining task, specifically the discovery task. The hypotheses of this research is that we would be able to define the characteristics of event logs which will be best fitted to be described with heuristic miner. This is beneficial for researchers with other case study to define whether heuristic miner will be best to analyze their event logs. The research is done separately for each case study. The final analysis will be done to get the conclusions of overall studies.

Keywords: heuristic miner, event log, discovery, conformance checking, enhancement

Introduction

Process mining is a new discipline which tries to understand and formulate information inside the real implementation of a process. The real implementation of a process is represented by event logs, which is usually automatically recorded by the information system. Process mining done in three types, which are: discovery, conformance checking, and enhancement. Although this research should focus only on discovery, we also implement the other two types, to complete the analysis and discussion.

Heuristic miner is a process mining algorithm which is said to be able to handle noises [5]. This algorithm also focuses on quantifying frequency, which makes heuristic miner can illustrate dependencies among complex events. This algorithm is interesting because some researchers had been implementing this algorithm in different cases.

In this research, heuristic miner is implemented to three different case studies to test the performance of heuristic miner in different types of event logs. The case studies are carefully selected to represent different types of event logs. Those three are considered enough to represent various event logs. The first and second case studies represent simple event logs with different characteristics. The third case study represents a complex event logs with many different traces.
The research is done separately for each case study, and the final analysis is done to get the conclusions of the overall study. The conclusion is expected to be useful for defining whether heuristic miner is suitable for another event logs.

Literature Review

Process Mining:

Process mining is a discipline as a combination of computation intelligence and data mining for process modelling and analysis [2]. Process mining is a method to gather information in a business process, which is represented in the form of event logs. The aims are to build a process model of event logs and to recommend a strategy to enhance business process model.
There are three types of process mining, which are *discovery*, *conformance checking*, and *enhancement* [2]. *Discovery* is the most common and basic type of process mining, which build a process model based on events dependencies in the event logs. *Conformance checking* focuses on evaluating process model resulted in the discovery with the event logs, to be used for further analysis. *Enhancement* is the type focuses on improving, renewing or creating a new process model based on the process model before, based on formerly unknown attributes, such as time, user, etc.
Figure 1 shows the relation of those three types of process mining.

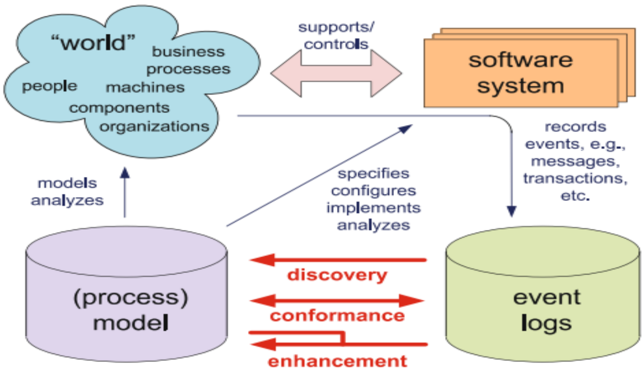


Figure 1: Process mining types [2]

When implementing process mining, relation between process model and event logs are important because they can support each type of process mining. There are three relations, which are: *Play-in, Play-out, and Replay* [2]. *Play-in* is a relation where process model is built based on generalization of the event log. *Play-out* is building event log based on a process model. *Replay* is the use of event log and process model as the input to repeat the use of event log for further analysis of the process model.

Heuristic Miner :

Heuristic Miner is an algorithm used specifically in the process model discovery. This algorithm focuses on calculating dependency frequency and traces of events in building a process model [1]. To build a process model, event logs need to be analyzed based on the dependency values of the activities.

The basic steps of heuristics miner are [5]:

1. Building dependency graph

Dependency graph is a model which represents dependency (causality) of events. To build a dependency graph, we need to build dependency matrix, length-one loop dependency, and length-two loop dependency.

2. Building causal matrix

Naturally, it is difficult to define whether a process is parallel of sequential to another process, just by analyzing the event log. Heuristic Miner builds causal matrix to represent the correct process model. There are two types of non-observal activities, which are AND andXOR. The AND type represents parallel activities, while the XOR type represents sequential activities.

When building a dependency graph, we can set some thresholds for events to be modeled [5], which are:

1. Dependency measure threshold: minimum value of dependency between events.
2. Positive observation threshold: minimum value of dependency frequency between events.
3. Relative to best threshold: minimum value of the difference between event dependency value with the maximum dependency value.
4. Length-one threshold: minimum value of same event dependency.
5. Length-two threshold: minimum value of looping pair event dependency.

Conformance Checking :

Conformance checking is the second type of process mining, which checks the process model resulted in the discovery type with the event logs. There are some approaches which can be used to do the conformance checking. In this research, we use Artificially Generated Negative Events (AGNEs) [7] using F-Measure to evaluate the process model.

It is based on the confusion matrix, as in Table 1.

Table 1: Confusion Matrix

		Actual	
		Positive	**Negative**
Prediction	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative(FN)	True Negative (TN)

Negative event is calculated to detect whether the process model is under-fitting or over-fitting, or not, based on the precision value [8]. Based on the confusion matrix, we can calculate precision, recall and f-measure of the model, with the formula:

$$Precision = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}$$

$$Recall \quad = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}}$$

$$F\text{-}Measure = \mathbf{2} \times \frac{\boldsymbol{precision \times recall}}{\boldsymbol{precision + recall}}$$

Conformance checking is done to define similarity and dissimilarity between process model behavior and data behavior. Another method is by calculating fitness value. Fitness calculates behavior proportion on event logs in the model [2]. Fitness is calculated by replaying every trace. We can use four measures, which are p (produced token), c (consumed token), m (missing token), and r (remaining token). The higher fitness value, the higher the similarity between model and activity. The fitness formula [2] is:

$$fitness(\sigma) = \frac{1}{2}\left(1 - \frac{m}{c}\right) + \frac{1}{2}\left(1 - \frac{r}{p}\right) \qquad \ldots (1)$$

Where:

σ : *trace*
p :*produced token*
c :*consumed token*
m :*missing token*
r :*remaining token*

Enhancement :

The idea of enhancement is to add, reduce, or fix a model process based on processes in the event logs [2]. There are two types of enhancement, which are:

1. *Repair*, which modifies process model to represent realities. If an activity in the model is sequential but actually can be done in any sequence, then the model needs to be fix based on the real sequence.
2. *Extension*, which is usually be done by adding new perspective in the process model. Example for the additional perspective is time perspective.

Research Methodology

This research is done experimentally by implementing heuristics miner in three different cases. Those three cases are chosen to represent a simple, medium, and complex event logs. The behavior and the result are then analyzed.

Case Studies :

The first case study is book procurement in a university. The parties involved in this process are: library unit, logistic unit, and the vendors. The process is started with the proposal of library unit to procure books. Logistic unit is the processing the proposal to find the suitable vendor. Figure 2 illustrates the flow of this process.

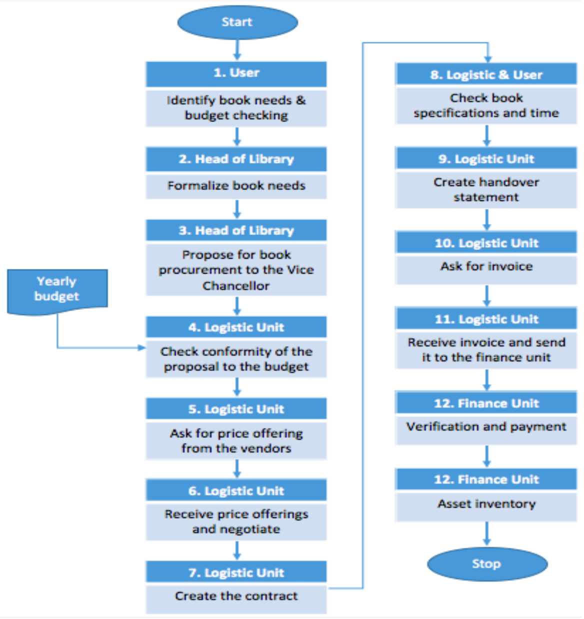


Figure 2: Flow of book procurement process

Figure 2 shows that there are 12 activities which are done sequentially in the book procurement process. This process is chosen as the first case study because of its simple and sequence flow.

The second case study is document control process at the University. The first activity of this process is submission of document creation/ update, and it is ended with document approval and distribution. Figure 3 illustrated the flow of this process.

The second case study is chosen because it is simple but having one conditional activity, which is "Revise on the draft", by the Process Owner. This case study has only three roles and eight (8) activity types. A simple case study allows us to understand the process better.

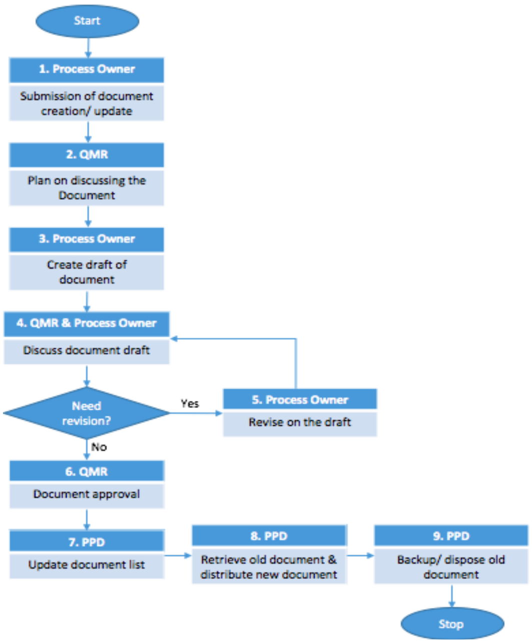


Figure 3: Flow of document control process

The third case study is a competition challenge of BPIC (Business Process Intelligence Challenge) 2014 [3], which is service desk processes of Rabobank. Among those three case studies, the third one is the most complicated. It is a real-life event log provided by the Rabobank Group ICT. The challenge is to design a (draft) predictive model, which can be used to implement in a BI environment. Different from the first and second case studies, we don't have a predefined process flow of this case.

Event Logs :

Event logs of those three case studies are analyzed as a step for understanding data of this research.

1. *Event logs of case study #1*

This event log has 14 event types, which are encoded into A-N. The users are: *library unit, logistic unit, supplier/ vendor, and finance unit*. Table 1 shows the example of the event logs.

Table 1: Example of book procurement event logs

Procure	Date	Activity	Actor
1	25/03/2004	A	Library
1	25/03/2004	B	Library
1	25/03/2004	D	Vice chancellor
1	25/03/2004	F	Logistic
1	25/03/2004	G	Logistic
1	26/03/2004	H	Supplier
2	04/08/2009	A	Library
2	04/08/2009	B	Library
2	04/08/2009	D	Vice chancellor
2	10/09/2009	E	Logistic
2	10/09/2009	F	Logistic
2	10/09/2009	G	Logistic

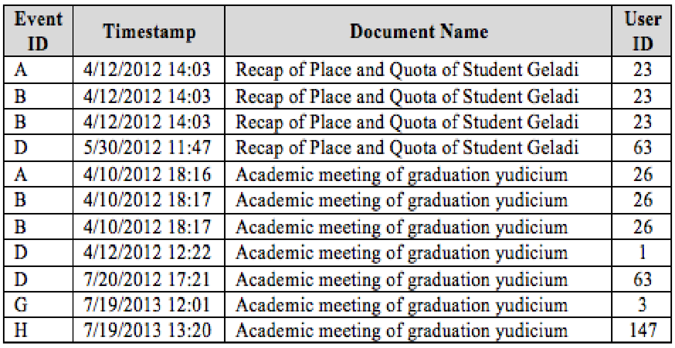
2. *Event logs of case study #2*

This event log has 7 event types, which are:

A = new document request
B = document upload
C = document request evaluate
D = document update
E = document request approval
F = undistributed document
G = distributed document
H = accept document

The users are: Process Owner, QMR (Quality Management Representative), and PPD (Data Officer). Table 2 shows example of the event logs.

Table 2: Example of document control event logs

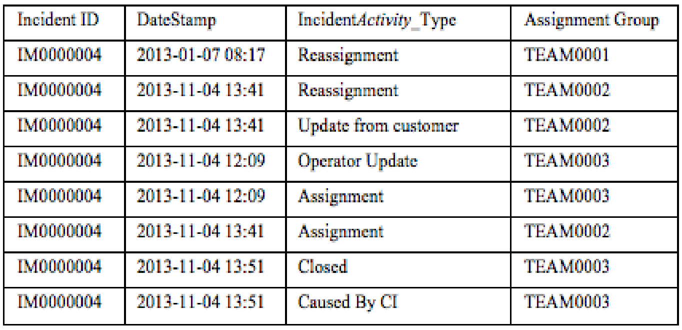


Event ID	Timestamp	Document Name	User ID
A	4/12/2012 14:03	Recap of Place and Quota of Student Geladi	23
B	4/12/2012 14:03	Recap of Place and Quota of Student Geladi	23
B	4/12/2012 14:03	Recap of Place and Quota of Student Geladi	23
D	5/30/2012 11:47	Recap of Place and Quota of Student Geladi	63
A	4/10/2012 18:16	Academic meeting of graduation yudicium	26
B	4/10/2012 18:17	Academic meeting of graduation yudicium	26
B	4/10/2012 18:17	Academic meeting of graduation yudicium	26
D	4/12/2012 12:22	Academic meeting of graduation yudicium	1
D	7/20/2012 17:21	Academic meeting of graduation yudicium	63
G	7/19/2013 12:01	Academic meeting of graduation yudicium	3
H	7/19/2013 13:20	Academic meeting of graduation yudicium	147

3. *Event logs of case study #3*

This event log has 7 activity types, which are: *Open*, *Assignment*, *Status change*, *Update*, *Reassignment*, *Operator update*, *Mail to customer*, *Quality Indicator Fixed*, *Caused by CI*,dan *Closed*. The example of the event logs is depicted in Tabel 3.

Table 3: Example of service desk event logs



Incident ID	DateStamp	Incident*Activity*_Type	Assignment Group
IM0000004	2013-01-07 08:17	Reassignment	TEAM0001
IM0000004	2013-11-04 13:41	Reassignment	TEAM0002
IM0000004	2013-11-04 13:41	Update from customer	TEAM0002
IM0000004	2013-11-04 12:09	Operator Update	TEAM0003
IM0000004	2013-11-04 12:09	Assignment	TEAM0003
IM0000004	2013-11-04 13:41	Assignment	TEAM0002
IM0000004	2013-11-04 13:51	Closed	TEAM0003
IM0000004	2013-11-04 13:51	Caused By CI	TEAM0003

Research Method :

Event logs of those three case studies are the input of this research. Those three event logs are then being used to analyze the performance of heuristics miner. The method of this research is illustrated in Figure 4.

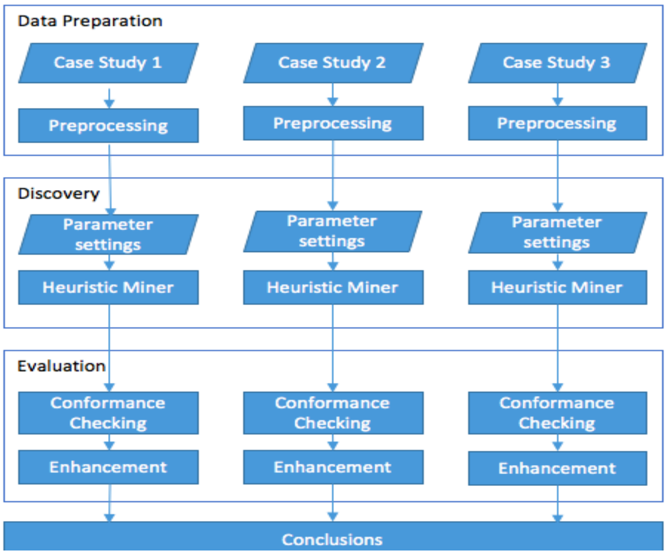


Figure 4: Research method

For each case study, we do the whole processes of process mining, which are preprocessing, discovery, conformance and enhancement. The results are then being analyzed to draw conclusions of this research.

Preprocessing is done for each event logs, ranging from data cleaning, data transformation, until the conversion into the XML format. Parameter settings in the discovery step is done to keep the heuristic miner performs on its best performance in different event logs.

Experimental Result

The results of the research are the heuristics net, conformance value and the enhancements of each case study. All of the results will be discussed in this part for each case study.

Experimental result of case study #1 :

Event logs of case study #1 is preprocessed by inconsistency cleaning, case id setting, and noise cleaning. The experiments also done by setting parameters into some values, including dependency threshold, positive observation threshold, relative to best threshold, loop length-one threshold, loop length-two threshold, and AND-XOR threshold. Model process of case study #1 is shown in Figure 5.

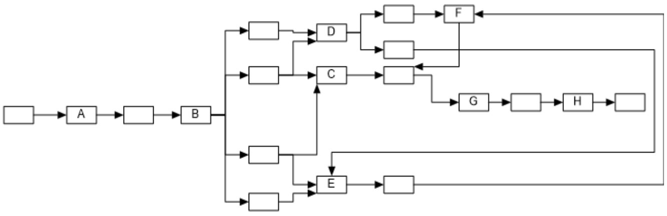


Figure 5: Process model of case study #1 [4]

Example of token replay for trace ABDFGHI can be calculated based on formula (1). Step by step of the token replay is shown on Fig. 6-13 [4]:

1. Before activity A (p=1, c=0, m=0, r=0)

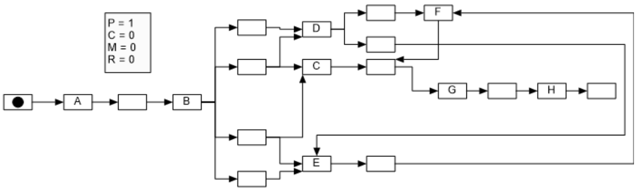


Figure 6: Replay before A

2. On activity A (p=2, c=1, m=0, r=0)

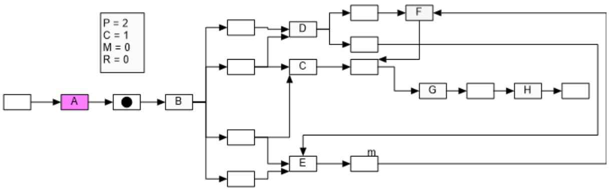


Figure 7: Replay on A

3. On activity B (p=6, c=2, m=0, r=0)

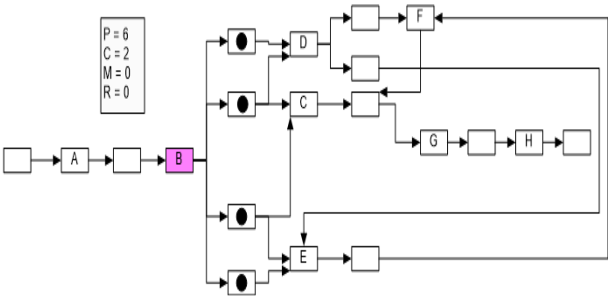


Figure 8: Replay on B

4. On activity D (p=8, c=4, m=0, r=0)

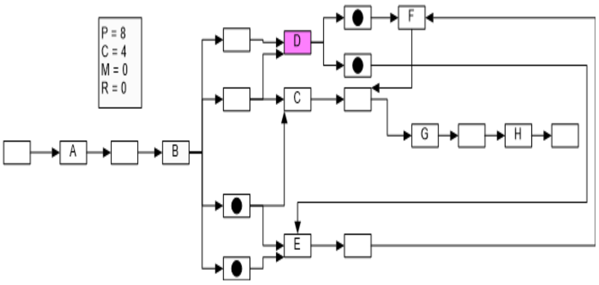


Figure 9: Replay on D

5. On activity F (p=9, c=6, m=1, r=0)

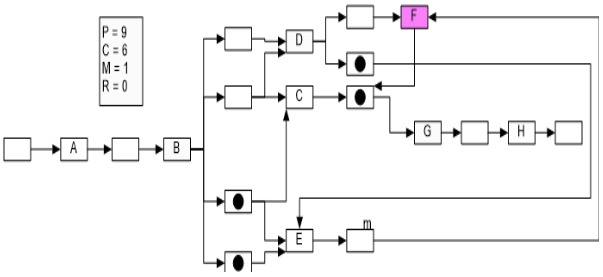


Figure 10: Replay on F

6. On activity G (p=10, c=7, m=1, r=0)

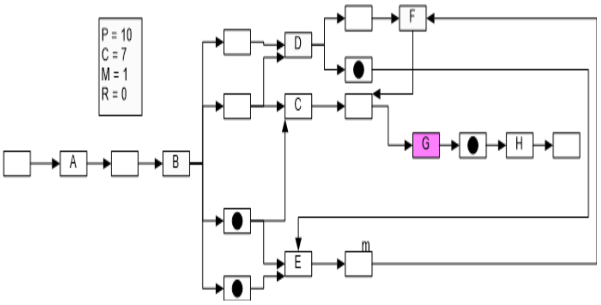


Figure 11: Replay on G

7. ***On activity H (p=11, c=8, m=1, r=0)***

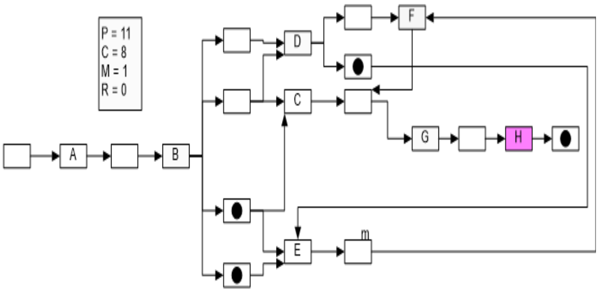


Figure 12: Replay on H

8. Finish (p=11, c=9, m=1, r=3)

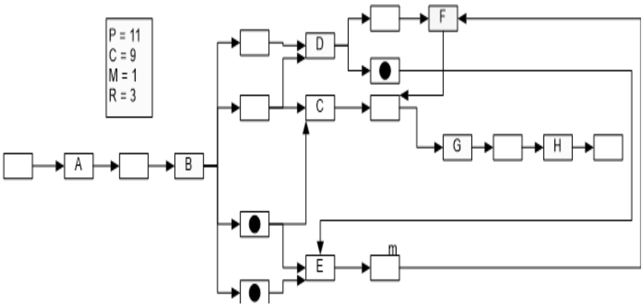


Figure 13: Replay at the finish

Fitness of the resulted model is calculated as follow:

$$fitness(1) = \frac{1}{2}\left(1 - \frac{1}{9}\right) + \frac{1}{2}\left(1 - \frac{3}{11}\right) = 0{,}44 + 0{,}36 = 0{,}8$$

Another experiment is by setting the threshold as: DM=0.8 (dependency measure), PO (positive observation)=12, and RTB (relative to best)=0.05.

In this case study, we build a recommended process model enhancement based on the patterns discovered after replaying all events in the event log. The enhanced process model is shown in Figure 14.

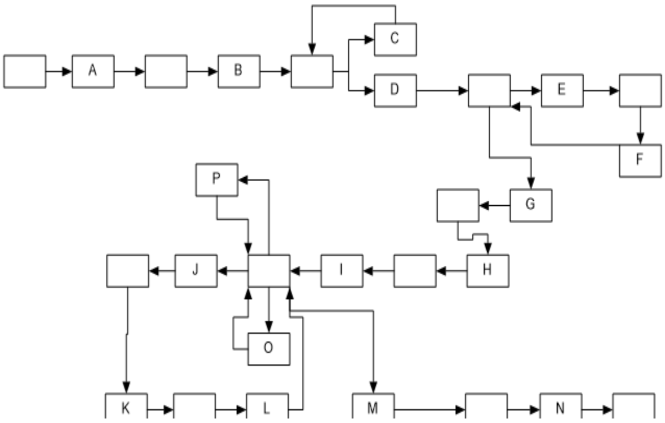


Figure 14: Enhanced process model of case study #1 [4]

Experimental result of case study #2 :
Heuristics net of case study #2 is shown in Figure 15.

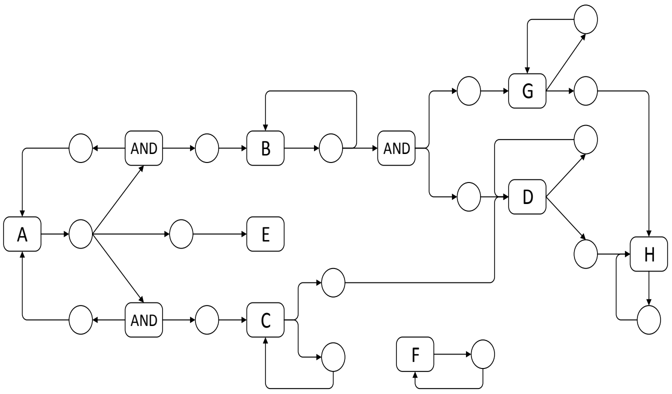


Figure 15: Heuristic net of case study #2 [6]

The best performance of heuristic net in case study #1 is measured by recall = 0.9701, precision = 0.8509, and f-measure = 0.9066. The threshold for this case is: DM=0.7, PO=10, and RBT=0.26.

The enhancement step of case study #1 shows some repairs which are need to be done:

1. New document without update should follow A→B→C→E→G→H.
2. New document with update should follow A→B→C→D→C→E→G→H.
3. Document revision without update should follow A→B→C→E→F→G→H.
4. Document revision with update should follow A→B→C→D→C→E→F→G→H.

The repaired process model of case study #2 is shown in Figure 16.

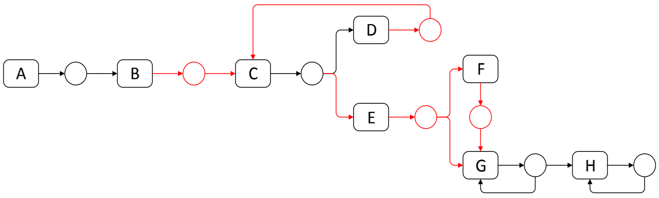


Figure 16: Repaired process model of case study #2 [6]

While for the extension, it is found that there are short bottleneck on activity A→B and long bottleneck on activities C→D, E→G, G→G and H→H. [6]

Experimental result of case study #3 :
Event logs of case study #3 is preprocessed by inconsistency cleaning, case id setting, and noise cleaning. The experiments also done by setting parameters into some values. Heuristic net of case study #3 is shown in Figure 17.

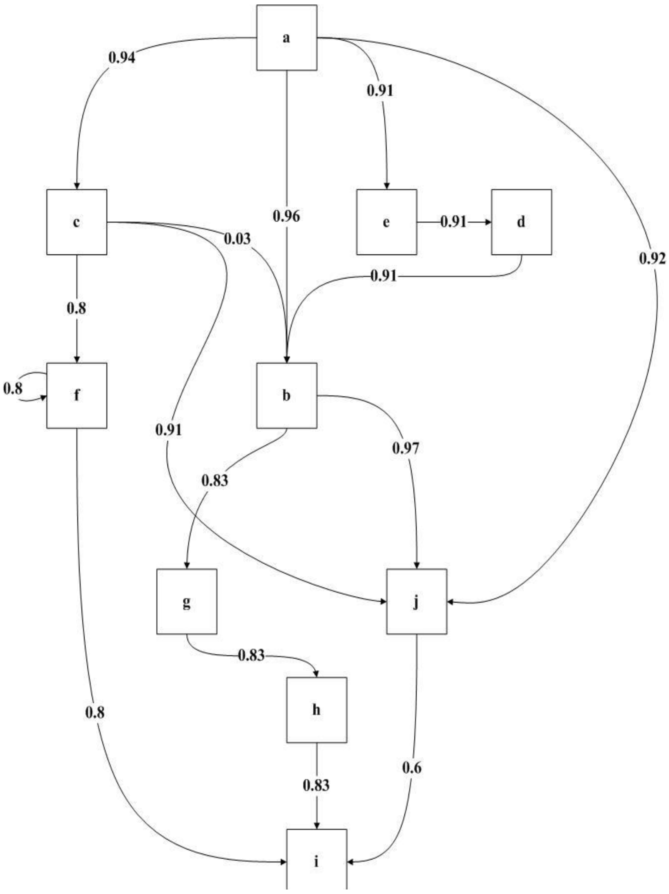


Figure 17: Heuristic net of case study #3 [8]

Setting parameters for this case study shows that DM and RBT are not significant for the fitness value. It means that we can set any value for both threshold and the fitness will not be affected. The best is POT = 1000. The best fitness for this case is 0.8494.
For the enhancement step, we add time and originator perspectives to be analyzed. It is found that heuristic miner reduces some connections, so that the organization should consider to reduces the real flows. The second result is that human capital couldn’t control work distribution among teams.

Conclusions

Conclusions of this research are based on experiments of those three cases. The result shows that:

1. Heuristic miner can be well implemented in different cases, with fitness value > 0.8. The performance can be seen on the biggest event logs we had, which is case study #3.
2. Performance of heuristic miner are depending on the parameter setting step. The best combination of the parameters should be found through trial-and-error and can be time consuming.
3. The best combination of parameter is depending on the event logs. Our three case studies shown that dependency measures and relative to best thresholds are less significant to the fitness on bigger event logs,

but the positive observation threshold is still significant.

Acknowledgments

This work was supported in part by the Competitive Research Grant of Indonesian Higher Education Ministry, 2015.

Special acknowledgement is mentioned by the first author for Mrs. Imelda Atastina, Mrs. Mira KaniaSabariah, Ms. HarinVeradistya, Mr. WildanKhalidy, and RendySetiadiMangunsong for the contribution in some parts of this research.

References

[1] StB Prof. Dr. NichGehrke and Michael Werner, Dipl.-Wirt.-Inf. "Process Mining". WISU, 2013.

[2] Wil MP. van der Aalst. 2011. "Process Mining: Discovery, Conformance and Enhancement of Business Process". Springer, German.

[3] The 10th International Workshop on Business Process Intelligence 2014, the 4th International Business Process Intelligence Challenge (BPIC 2014). Insights from the Analysis of Rabobank Service Desk Processes.

[4] HarinVeradistya Maharani, Angelina Prima Kurniati, Imelda Atastina. Process Mining on Book Procurement Process using Heuristic Miner Algorithm (Case Study: Telkom University Library). E-Proceeding of Engineering: Vol. 2 No. 1 April 2015.

[5] A.J.M.M. Weijters, W.M.P van der Aalst, and A.K. Alves de Medeiros. 2013. "Process Mining with TheHeuristicsMiner Algorithm".

[6] WildanKhalidy, Angelina Prima Kurniati, Imelda Atastina. Process Mining Implementation on Document Control Procedure using Heuristic Miner Algorithm (Case Study: Telkom Engineering School). E-Proceeding of Engineering: Vol. 2 No. 1 April 2015.

[7] D.M.J.V., B.B. StijnGoedertier. "Robust Process Discovery with Artificial Negative Events".

[8] RendySetiadiMangunsong, Angelina Prima Kurniati, Mira KaniaSabariah. Analysis and Implementation of Process Mining using Heuristic Miner Algorithm (Case Study: Event Logs of Rabobank Group ICT Netherlands). E-Proceeding of Engineering: Vol. 2 No. 1 April 2015.