Optimizing Process Discovery Quality Criteria and Model Measurements using Receiver Operating Characteristic Analysis and Infrequent Inductive Miner

R. Budiraharjo[1, 2]
[1]*Department of Informatics*
Institut Teknologi Sepuluh Nopember, Surabaya
[2]*Department of Information Systems*
Institut Teknologi Nasional, Bandung
Indonesia
budiraharjo.207025@mhs.its.ac.id

Riyanarto Sarno[1]
[1]*Department of Informatics*
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
riyanarto@if.its.ac.id

Hanung Nindito Prasetyo[1, 2]
[1]*Department of Informatics*
Institut Teknologi Sepuluh Nopember, Surabaya
[2]*Department of Information Systems Diploma*
Telkom University, Bandung
Indonesia
hanung.207025@mhs.its.ac.id

Kelly Rossa Sungkono[1]
[1]*Department of Informatics*
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
kelly@its.ac.id

***Abstract* -- Generating process models that reflect close behavioral resemblance to the actual process Standard Operating Procedure (SOP) in process mining can be challenging without taking the four quality criteria of process discovery into account. The four quality criteria, i.e. fitness, precision, generalization, and simplicity, should be well balanced in order to produce proper process models which are aligned to the real-life executions. This paper proposes a method to optimize process discovery quality criteria (PDQC) by implementing different thresholds and analyzing calculation results using Receiver Operating Characteristic (ROC) curve and Infrequent Inductive Miner algorithm. This paper sets up two experiments with different scenarios to measure the calculations of quality criteria and the quality of generated models. The experiments compare two SOPs to the process models discovered by Infrequent Inductive Miner algorithm; hence the SOPs serve as references to determine the generated models quality. The purpose of applying two different scenarios in the experiments is to discover how well the Infrequent Inductive Miner thresholds can produce predictive models under these two different scenarios circumstances. This paper has been successful in predicting the best-fit model in reference to the SOPs by optimizing the four quality criteria of process discovery using ROC thresholds settings and by using infrequent inductive miner algorithm for models generation, and also in improving the accuracy of models measurements. The accuracy rate of the prediction model from Experiment 1 is 83%, while Experiment 2 yields an accuracy rate of 88%. The most optimal threshold settings to generate the best model in this paper are threshold 0.4 in Experiment 1 and threshold 0.5 in Experiment 2.**

***Keywords* -- Process mining, process discovery, quality criteria, ROC curve, Infrequent Inductive Miner, standard operating procedure, event log**

I. Introduction

The purpose of process discovery, which is a sub study of process mining, is to generate process models from event logs or event data derived from information systems [1]. The quality of generated process models in process discovery largely depends on the reliability of event logs and algorithm used for extraction. Measuring the quality of process models should be easier and more relevant when SOPs that regulate how processes should be performed are available and used as references[1][2].

Four dimensions of quality criteria should be taken into account when generating models to optimize the quality of process models. These criteria are: Fitness, precision, generalization, and simplicity [1][2][3]. The fitness dimension concerns with how traces in event logs can be replayed by the models [1]. However, high fitness score would include all traces, which means that the models also accommodate traces which actually do not conform to actual procedures [4]. Precision is related to how far a model would allow behaviors in its structure. Generalization has a role in making sure that a model does not strictly limit behavior to what is seen in the event log, while simplicity dimension aims to prevent a model from displaying too complex behavior [1][4][5][6]. More detailed explanations about these four competing criteria will be discussed in later part of this paper.

In order to produce process models that closely represent their real-life SOPs, the four criteria should be well balanced to prevent the resulted models from *underfitting* or *overfitting* [1]. Balancing the four quality criteria for process discovery can be very challenging as those criteria are most often not supplementary to each other [2].

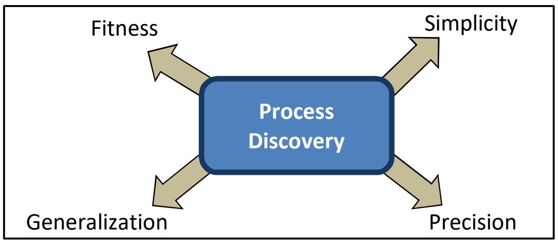

Fig.1. The four competing criteria of process discovery [1]

There are quite a few studies that have proposed methods in managing and balancing these criteria. Most of the proposed methods are limited to focus on compensating two of the four competing criteria. One such example, Seppe K.L.M vanden Broucke, et al. in [7] proposed a method for determining precision and generalization of process models using weighted artificial negative events. The study has been successful in giving contributions in forms of techniques in dealing with incomplete logs, a new generalization metrics, and their application on Petri net models. These new techniques and metrics have been made available in ProM [7].

Another example of such studies, Anja F. Syring, et. al in [8] formulate and evaluate several conformance propositions. The study revealed wide differences between propositions that were evaluated. Many of the requirements that were evaluated in the study did not meet by contemporary conformance measures [8]. Most of, if not all, the propositions discussed in the study concentrated on precision and generalization dimensions.

In this particular paper, there are several implementations and their expected outcomes resulted in from conducting the methodology. Firstly, this paper proposes an approach for balancing all four of quality criteria in process discovery. The proposed method uses R.O.C curve as a tool for conducting the quality criteria measurements. Secondly, this paper discusses techniques for determining optimal threshold for generating the best models in relation to the SOPs. And, lastly, this paper evaluates the performance of Infrequent Inductive Miner algorithm in ProM in generating process models under different experiments scenarios and threshold settings.

II. PRELIMINARIES

In this section of this paper, basic notations and background concepts are introduced in relation to process mining, process discovery, R.O.C., and the Inductive Miner algorithm fields of study areas.

A. Process Mining

Process mining is a relatively new field of study that concentrates on discovering, evaluating, and improving business processes. Most researchers put this young discipline between machine learning and data mining [9]. Process mining activities can be simply described as generating knowledge of how business processes are actually carried out by extracting execution models from event logs that are available in information systems [9][2].

Scholarly researchers agree that there are three main classifications of process mining techniques, which they are:

1) Process Discovery: This technique focuses on creating new process models based on event logs data extractions. Further explanations for process discovery will be discussed later in this paper [1].

2) Conformance Analysis: This technique is primarily used when there are already procedures or priori models of certain business processes. The goal of this activity, as the name suggests, is to evaluate whether or not processes are carried out following procedures. Conformance analysis can also be used to locate problems in existing models and to improve the models based on the findings. Finding bottlenecks is one of the most regularly performed activities from this technique, for example [1].

3) Performance Mining or Process Enhancement: This technique is actually the extension of conformance analysis technique where procedures or priori models that have already been in place are required [1]. Performance mining concentrates on other aspects of the reviewed models such as time of processing, waiting times, cycle periods, and financial or non-financial costs. Thus, the goal of this activity is not to check process conformance but to improve the processes and to enhance the existing procedures.

B. Process Discovery

Process discovery is the first classification and some might argue is the most challenging activity in process mining domain. The main idea of this activity is to capture behavior of a system by extracting its event log. The outcomes from the activity are model processes [1]. To generate such models, various algorithms are used to map the logs into representative models.

The requirement that the generated models from the algorithms are “representative” does not give many details about how the implementations should be carried out. To achieve the “representativeness”, there are four quality criteria that should be taken into account when generating models [1]. Namely, the criteria are:

1) Fitness: In general, fitness dimension accommodates behaviors seen in the event logs to be replayed in the generated models. The measurement for this criterion is valued ranging from 0 to 1 [4][1]. However, it is useful to keep in mind that when a model has a very high fitness value, this would also mean the model maps all traces including noises and incompleteness.

Fitness dimension quality (Q_f) can be measured by averaging *Parsing Measure* (*PM*) and *Continuous Parsing Measure* (*CPM*) [4]. The mathematical formulas for fitness quality measurements can be expressed as follow:

$$\mathrm{PM} = \frac{c}{t} \quad (1)$$

$$\mathrm{CPM} = \frac{1}{2}\frac{(e-m)}{e} + \frac{1}{2}\frac{(e-r)}{e} \quad (2)$$

$$Q_f = \frac{CPM+PM}{2} \quad (3)$$

where:

c : number of traces that are mapped in the generated process model.

t : total number of traces in an event log

m : total number of activities that are not included in the process model

r : number of remaining activities in an event log after a model is generated

e : total number of activities in an event log.

2) Precision: This dimension can be treated as one of the aspects for evaluating a generated model. Precision criterion focuses on how a model should represent activities that are captured in an event log [1]. Precision is, in the same sense, related to *overfitting* notation in data mining context. A model is overfit when it specifically replay all of the traces in an event log. As in fitness dimension, precision quality measurement is also valued ranging from 0 to 1 [4].

Confusion matrices are required when calculating precision quality. The mathematical formula to perform precision calculation (Q_p) can be expressed as follow:

$$Q_p = \frac{tp}{p'} \tag{4}$$

where:

tp: number of true positives in an event log traces included in a process model

p' : total of true positives and false positives of all the traces included in a process model.

3) Generalization: This dimension denotes that a generated model needs to generalize in order to accommodate behaviors not seen in an event log [1]. Generalization scores ranging from 0 to 1 wherein scores closer to 1 will generate underfit models [4]. Generalization can be calculated using the following equation.

$$Q_g = 1 - \frac{\sum_1^t (\sqrt{e})^{-1}}{o} \tag{5}$$

where:

o : the number of operator nodes in a process tree

t : operator nodes implemented in an event log

e : the number of operator nodes implemented in traces in an event log

4) Simplicity: This dimension of process discovery embraces the principle that the simpler a model, the better [1]. The simplicity measurement in a model can be calculated by comparing the size of the process tree of a model to the number of activities in an event log [4].

The calculation for simplicity dimension can be performed using the following equation.

$$Q_s = 1 - \frac{r + a}{l + v} \tag{6}$$

where:

r : the number of activity variants in an event log that redundantly appear in the process tree

a : the number of activity variants in an event log that do not appear in the process tree

l : the number of leaf nodes in a process tree

v : the number of activity variants in an event log

C. Receiver Operating Characteristic (ROC)

ROC is a graphical curve that demonstrates the accuracy of a classifier system as its threshold is set to various position. The ROC curve in general is created by plotting the true positive rate (TPR) against the false positive rate (FPR) [10].

In performing its calculation, ROC typically utilizes confusion matrix which is a kind of contingency table with actual and predicted dimensions. This matrix is used to calculate the values of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) in any given threshold setting [10]. The calculation results from the confusion matrices are then further calculated to discover the value of TPR, FPR, true negative rate (TNR), and false negative rate (FNR). The next step is normally setting up an ROC plot curve from the various thresholds calculations from earlier stage. The ROC plot helps us compare classifiers (from the thresholds settings) and determining the classifier that is closest to (0,1) and the furthest from TPR=FPR [11]. In simpler explanation, this means the best predictor would be located near or at the top left corner of the ROC curve.

The following equations are used in this paper to create and analyze ROC curves [10][11][12][13].

$$\text{TPR} = \frac{TP}{P} = \frac{TP}{TP+FN} = 1 - FNR \tag{7}$$

$$\text{TNR} = \frac{TN}{N} = \frac{TN}{TN+FP} = 1 - FPR \tag{8}$$

$$\text{FPR} = \frac{FP}{N} = \frac{FP}{FP+TN} = 1 - TNR \tag{9}$$

$$\text{FNR} = \frac{FN}{P} = \frac{FN}{FN+TP} = 1 - TPR \tag{10}$$

$$\text{ACC} = \frac{TP+TN}{P+N} = \frac{TP+TN}{TP+TN+FP+FN} \tag{11}$$

where:

P : condition positive

N : condition negative

ACC : accuracy

D. Infrequent Inductive Miner Algorithm

Infrequent Inductive Miner is a variant of Inductive Miner algorithm. Inductive miner is a discovery approach to create a Process Tree for a given event log [14]. This algorithm offers advantages such as the discovered models always correspond to sound, block-structured workflow net (WF) systems [15]. This algorithm is embedded as a plug in in ProM.

Infrequent Inductive Miner algorithm enables users to set various thresholds in process discovery in accordance of necessity [16]. This, hence, is a suitable tool to be used for testing Experiment 1 and Experiment 2 in this paper.

III. Experiments

This paper uses two experiments scenarios, which are:

1) Experiment 1: Uses 12 traces in which the traces do not represent all of the possible paths of its SOP process model. The SOP designed for Experiment 1 has complex behavior containing a short loop and invisible tasks. The data set used to test this scenario is not normalized which means noises such as uncompleted tasks are calculated.

2) Experiment 2: Uses 32 traces accommodating all possible paths of its SOP. The SOP used for Experiment 2 is designed in a simpler manner, compared to that of Experiment 1, containing only invisible tasks without loops. The data set used in Experiment 2 was normalized before being processed in this experiment. The normalization aims to eliminate noises in the event log.

The goal of using two different experiment scenarios is to find out the soundness of process discovery under these two different circumstances using PDQC optimization method proposed in this paper. Both Experiment 1 and Experiment 2 deploy 11 models to be simulated. Each model simulates different threshold settings, ranging from 0 to 1 threshold set ups. Fitness, precision, generalization, and simplicity scores are all calculated in every model.

PDQC optimization method proposes 10 steps in running these experiments. The steps are in this following order: First, SOPs are designed to be used as references in both scenarios. Second, synthetic event logs are created to be used for simulations in these experiments. Third, the dataset is normalized before it is used for further simulation (this third step is exclusively for Experiment 2 scenario). Fourth, conformance tables are made to mark the traces conformity status. Fifth, the conformance tables are scatter plotted into threshold curves. Sixth, fitness (F), precision (P), generalization (G), and simplicity (S) scores are calculated for each of the 11 threshold simulation models using (1), (2), (3), (4), (5), and (6). Seventh, calculation results from step sixth is then processed using (7), (8), (9), and (10) to figure out the TPR, FPR, TNR, and FNR. Eighth, the calculation results are analyzed and mapped into ROC curves. Ninth, the prediction model accuracy is calculated using (11) to measure the quality of the prediction model. And tenth, the most optimum thresholds are selected to generate the closest model in reference to the SOPs.

The SOPs, conformance tables, threshold curves, ROC curves, and ProM 6 are the primary tools used in the experiments. However, there are also other tools that are indispensible in conducting the experiments. The complete sets of tools used in this paper are:

a) Standard Operating Procedure (SOP): An SOP can be described as a sequence of instructions that is designed to be a reference of how routine operations should be carried out. The SOPs used in this paper were designed in different behaviors to test the quality of proposed discovery method.

b) Synthetic Event Logs: A synthetic event log is an artificial event log which is normally made to fill specific needs, usually for demonstrations and simulations purposes. This paper uses two synthetic event logs to be adapted to Experiment 1 and Experiment 2.

c) Conformance Table: A conformance table is simply a table that illustrates conformity level of subjects towards a particular standard. In this paper, the subjects are the traces from the synthetic event logs and the standards are the SOPs designed for Experiment 1 and Experiment 2. This paper uses two conformance tables, one for each experiment.

d) Scatter Plot Graph: A scatter plot is a diagram using Cartesian coordinates to display data with two variables. In performing ROC analyses, two scatter plot diagrams are used to help with the measurements in this paper. Fig. 2 shows an example of a scatter plot diagram used in this paper.

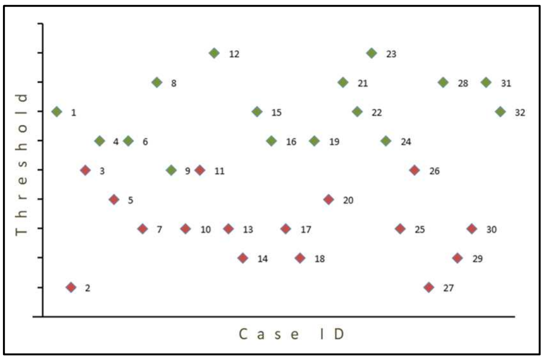

Fig.2. Scatter plot graph used for Experiment 2 in this paper

e) Confusion Matrix: A confusion matrix is a contingency table with actual and predicted dimensions. In general this table is used to report the values of TP, FP, TN, and FN. Confusion matrices offer more detailed analysis compared to other methods. In this paper, a confusion matrix is used in each 11 models in both Experiment 1 and Experiment 2. In total, there are 22 confusion matrices used to perform ROC calculations in this paper.

f) Process Tree: A process tree is a schematic diagram of processes. In simple explanation, a process tree shows how a subject related to other subjects in a hierarchical model. Fig. 3 shows an example of a process tree that is used in this paper.

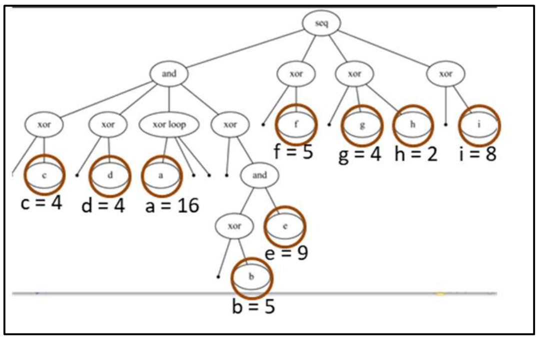

Fig.3. The process tree of ROC threshold 0.1 setting in Experiment 1

g) ProM 6: ProM 6 is an open source application of process mining discipline that is used to generate process models from event logs. ProM 6 is an extensible framework that offers numbers of plug-ins for process mining techniques. This paper uses ProM 6 to run the Infrequent Inductive Miner algorithm.

IV. Results and Discussion

The results and analysis for both Experiment 1 and experiment 2 can be described as follows.

1) Experiment 1: Following the 10 steps (minus the third step) of PDQC optimization method, Experiment 1 has resulted in simulations that some of the most illustrating steps are discussed in this section. The discussion is as follow.

SOP used for Experiment 1 can be observed in Fig 4.

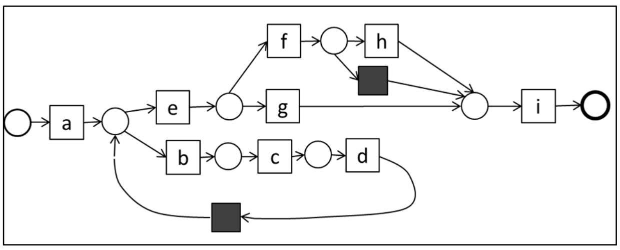

Fig.4. Standard Operating Procedure designed for Experiment 1

Table I in the next page shows conformance status of the traces of the synthetic event log created for Experiment 1.

TABLE I: EXPERIMENT 1 CONFORMANCE TABLE

Case ID	*Trace*	*Number of Activities*	*Conformance*
1	*(a, i)*	2	0
2	*(a, e, f, i)*	4	1
3	*(a, b, c, d, a, e, f, i)*	8	1
4	*(a)*	1	0
5	*(a, g, i)*	3	0
6	*(a, e, f, h, i)*	5	1
7	*(a, b, c, d, a, e, f, h, i)*	9	1
8	*(a, e, f, g)*	4	0
9	*(a, b, c, d, a, e)*	6	0
10	*(a, b, c, d, a, e, g, i)*	8	1
11	*(a, e, b)*	3	0
12	*(a, e, g, i)*	4	1

Table II shows calculations results for F, P, G, and S criteria; the numbers of TP, FP, TN, and FN; also the scores of TPR, FPR, TNR, and FNR for each threshold (Thsld) in Experiment 1.

TABLE II: PREDICTION MODEL OF EXPERIMENT 1

Prediction Model												
Thsld	*TP*	*FP*	*TN*	*FN*	*TPR*	*FPR*	*TNR*	*FNR*	*F*	*P*	*G*	*S*
0.0	6	6	0	0	1.0	1.0	0.0	0.0	1.00	1.00	0.55	0.50
0.1	6	6	0	0	1.0	1.0	0.0	0.0	1.00	1.00	0.55	0.50
0.2	6	5	1	0	1.0	0.8	0.2	0.0	0.95	1.00	0.55	0.50
0.3	6	4	2	0	1.0	0.7	0.3	0.0	0.90	1.00	0.54	0.50
0.4	6	2	4	0	1.0	0.3	0.7	0.0	0.79	1.00	0.52	0.50
0.5	4	1	5	2	0.7	0.2	0.8	0.3	0.62	0.67	0.44	0.56
0.6	3	1	5	3	0.5	0.2	0.8	0.5	0.55	0.50	0.37	0.61
0.7	3	0	6	3	0.5	0.0	1.0	0.5	0.48	0.50	0.33	0.61
0.8	3	0	6	3	0.5	0.0	1.0	0.5	0.48	0.50	0.33	0.61
0.9	1	0	6	5	0.2	0.0	1.0	0.8	0.32	0.17	0.14	0.94
1.0	0	0	6	6	0.0	0.0	1.0	1.0	0.00	0.00	0.00	0.00

Fig. 5 shows calculation results of the quality criteria for each of 11 models deployed in Experiment 1.

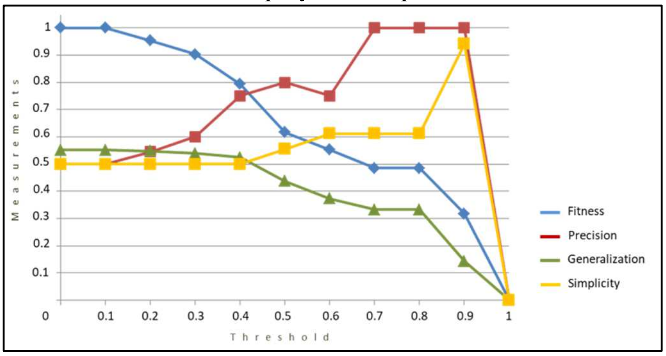

Fig. 5. Quality criteria chart for Experiment 1

Process models generated from ProM 6 using Infrequent Inductive Miner algorithm that show underfit, optimal fit, and overfit models resulted from different threshold settings in Experiment 1 can be seen in Fig. 6, Fig. 7, and Fig. 8.

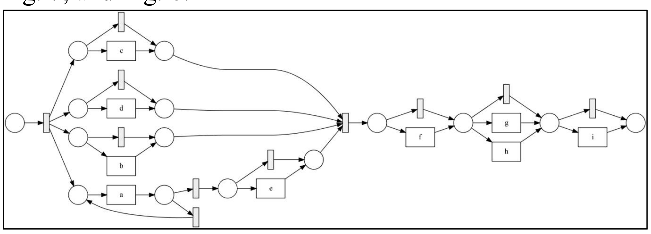
Fig. 6. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.1 threshold setting results in an underfit model

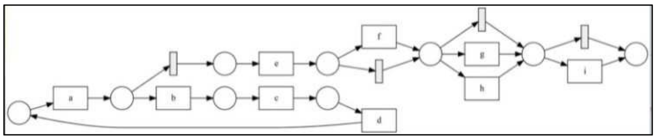
Fig. 7. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.4 threshold setting results in an optimal fit model

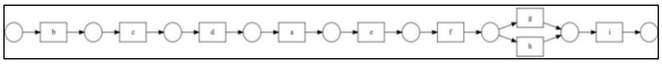
Fig. 8. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.7 threshold setting results in an overfit model

At this stage, the prediction model from Table 2 is then mapped into an ROC curve by plotting TPR against FPR, as seen in Fig. 9.

Fig. 9 suggests that threshold 0.4 serves as the best predictor among other predictors for it is located in the furthest point from the diagonal random classifier line in the middle of the curve. Threshold 0.4 represents 100% of TN and 30% of FN, thus offers 83% accuracy rate (calculated using (11) → $\frac{TP+TN}{TP+TN+FP+FN} = \frac{6+4}{6+2+4+0} = 0.83$) for Experiment 1 scenario. This statement aligns with Fig. 7 where the process model is generated using Infrequent Inductive Miner algorithm in ProM 6 with 0.4 threshold setting. The model shows closest resemblance to the SOP used as reference for Experiment 1.

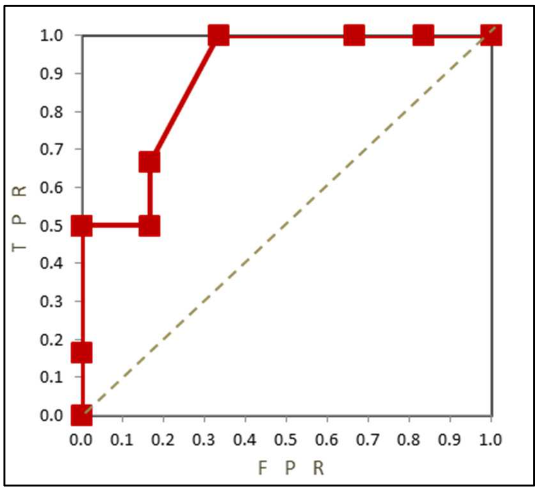

Fig. 9. ROC curve for Experiment 1

2) Experiment 2: Employing all of the 10 steps of PDQC method, the simulation results of Experiment 2 will be discussed in this section.

The SOP for Experiment 2 is as seen in Fig. 10.

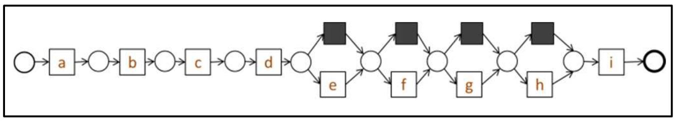

Fig.10. Standard Operating Procedure used for Experiment 2

Conformance table for Experiment 2 is as described in Table III.

TABLE III: EXPERIMENT 2 CONFORMANCE TABLE

Case ID	*Trace*	*Number of Activities*	*Conformance*
1	*(a, b, c, d, e, f, i)*	7	1
2	*(a)*	1	0
3	*(a, b, c, d, e)*	5	0
4	*(a, b, c, d, e, i)*	6	1
5	*(a, b, c, d)*	4	0
6	*(a, b, c, d, f, i)*	6	1
7	*(a, b, c)*	3	0
8	*(a, b, c, d, f, g, h, i)*	8	1
9	*(a, b, c, d, i)*	5	1
10	*(a, b, c)*	3	0
11	*(a, b, c, d, f)*	5	0
12	*(a, b, c, d, e, f, g, h, i)*	9	1
13	*(a, b, c)*	3	0
14	*(a, b)*	2	0
15	*(a, b, c, d, f, g, i)*	7	1
16	*(a, b, c, d, e, i)*	6	1
17	*(a, b, c)*	3	0
18	*(a, b)*	2	0
19	*(a, b, c, d, f, i)*	6	1
20	*(a, b, c, d)*	4	0
21	*(a, b, c, d, e, f, g, i)*	8	1
22	*(a, b, c, d, f, g, i)*	7	1
23	*(a, b, c, d, e, f, g, h, i)*	9	1
24	*(a, b, c, d, e, i)*	6	1
25	*(a, b, c)*	3	0
26	*(a, b, c, d, f)*	5	0
27	*(a)*	1	0
28	*(a, b, c, d, e, f, g, i)*	8	1
29	*(a, b)*	2	0
30	*(a, b, c)*	3	0
31	*(a, b, c, d, f, g, h, i)*	8	1
32	*(a, b, c, d, e, f, i)*	7	1

The dataset used for Experiment 2 was tested for normality using *Kolmogorov–Smirnov* before processed into calculation phase. The detailed result of normality test can be explained as follow:

H0: Normal data distribution
H1: Not normal data distribution
Significance: 0.094
Indicator: P Value > 0.05, then H0 is accepted.
Conclusion: The data is normally distributed

Table IV describes scores for F, P, G, and S criteria; the numbers of TP, FP, TN, and FN; and the scores of TPR, FPR, TNR, and FNR for each threshold prediction model in Experiment 2.

TABLE IV: PREDICTION MODEL OF EXPERIMENT 2

Prediction Model												
Thsld	***TP***	***FP***	***TN***	***FN***	***TPR***	***FPR***	***TNR***	***FNR***	***F***	***P***	***G***	***S***
0.0	16	16	0	0	1.0	1.0	0.0	0.0	1.00	1.00	0.73	0.50
0.1	16	16	0	0	1.0	1.0	0.0	0.0	1.00	1.00	0.73	0.50
0.2	16	14	2	0	1.0	0.9	0.1	0.0	0.97	1.00	0.73	0.50
0.3	16	11	5	0	1.0	0.7	0.3	0.0	0.91	1.00	0.72	0.50
0.4	16	5	11	0	1.0	0.3	0.7	0.0	0.78	1.00	0.72	0.50
0.5	16	3	13	0	1.0	0.2	0.8	0.0	0.74	1.00	0.71	0.50
0.6	15	0	16	1	0.9	0.0	1.0	0.1	0.65	0.94	0.69	0.50
0.7	10	0	16	6	0.6	0.0	1.0	0.4	0.53	0.63	0.65	0.50
0.8	6	0	16	10	0.4	0.0	1.0	0.6	0.42	0.38	0.57	0.50
0.9	2	0	16	14	0.1	0.0	1.0	0.9	0.31	0.13	0.29	0.50
1.0	0	0	16	16	0.0	0.0	1.0	1.0	0.00	0.00	0.00	0.00

Fig. 12 describes results of the quality criteria calculation for all of the 11 models used in Experiment 2.

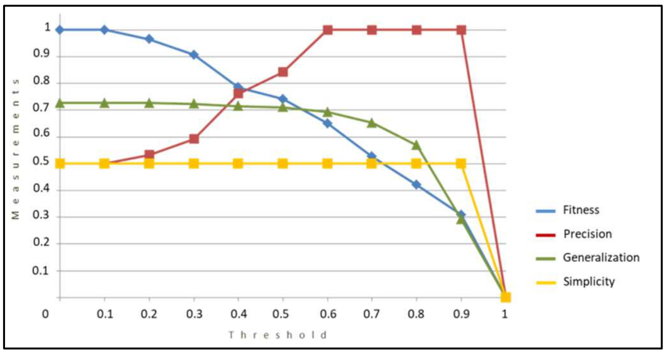

Fig. 12. Quality criteria chart for Experiment 2

Both Fig. 5 and Fig. 12 suggest that F, P, and G quality criteria count down while S criterion tends to score up as the threshold is raised up.

Process models generated from ProM 6 using Infrequent Inductive Miner algorithm that show underfit, optimal fit, and overfit models resulted from different threshold settings in Experiment 2 can be seen in Fig. 13, Fig. 14, and Fig 15 respectively.

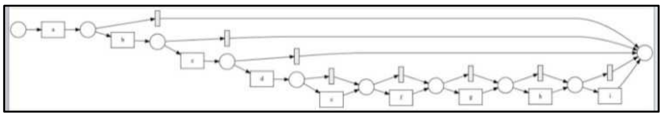

Fig. 13. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.1 threshold setting in Experiment 2 results in an underfit model

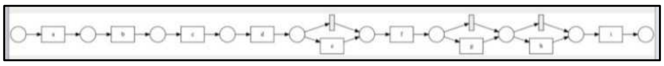

Fig. 14. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.5 threshold setting in Experiment 2 results in an optimal fit model

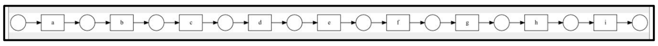

Fig. 15. Process model generation using Infrequent Inductive Miner algorithm in ProM 6 with 0.8 threshold setting in Experiment 2 results in an overfit model

As implemented in Experiment 1, the next stage of demonstration in Experiment 2 will describe how the prediction model is mapped into an ROC curve by plotting TPR against FPR for further analysis, as shown in Fig. 16.

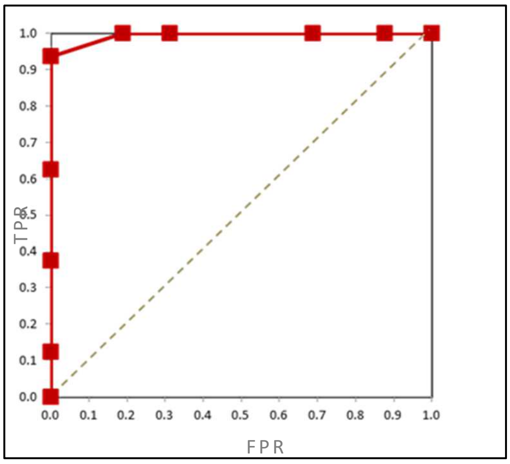

Fig. 16. ROC curve for Experiment 2

Fig. 16 describes that threshold 0.5 is the most optimal predictor among other predictors for Experiment 2 for it represents 100% of TN and 20% of FN, thus offers 88% accuracy rate (calculated using (11) → $\frac{TP+TN}{TP+TN+FP+FN} = \frac{16+13}{16+3+13+0} = 0.88$) for this scenario. This description is consistent with Fig. 14 which depicts a process model that shows very close resemblance to the SOP used as reference for Experiment 2. The process model in Fig. 14 is generated using Infrequent Inductive Miner algorithm in ProM 6 with 0.5 threshold setting.

V. Conclusion

PDQC optimization method using ROC analysis and Infrequent Inductive Miner algorithm proposed in this paper has been successful in optimizing process discovery to generate the best-fit models in reference to the SOPs used in the experiments. The method proposed in this paper has also improved the measurements accuracy of the prediction models.

The various threshold settings in the scatter plot curve in all 11 experiment models of both experiments have concluded that: In Experiment 1, increasing value of threshold results in descending values of Fitness, Precision, Generalization and ascending value of Simplicity; In Experiment 2, increasing value of threshold results in descending values of Fitness, Precision, Generalization, and stagnant value of Simplicity. The unchanged value of Simplicity in Experiment 2 is due to the event log of Experiment 2 containing significantly more traces than the number of traces in Experiment 1 event log. In general, the more traces in an event log the more likely all activity variants in the event log will appear redundant in the process tree, hence the Simplicity value is not affected.

The comparison between two different scenarios of Experiment 1 and Experiment 2 has concluded that Experiment 2 scenario has provided better and more accurate prediction for process discovery. The accuracy rate of the prediction model from Experiment 1 is 83%, while Experiment 2 yields an accuracy rate of 88%. The most optimal threshold settings to generate the best model in this paper are threshold 0.4 in Experiment 1 and threshold 0.5 in Experiment 2.

References

[1] W. M. P. van der Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. 2011.

[2] W. Van der Aalst, *Process mining: Data science in action*. 2016.

[3] S. a White and I. B. M. Corp, “Process Modeling Notations and Workflow Patterns,” *Business*, 2004.

[4] R. Sarno and K. R. Sungkono, “Coupled Hidden Markov Model for Process Discovery of Non-Free Choice and Invisible Prime Tasks,” 2017, doi: 10.1016/j.procs.2017.12.139.

[5] C. S. Wahyuni, K. R. Sungkono, and R. Sarno, “Novel parallel business process similarity methods based on weighted-tree declarative pattern models,” *Int. J. Intell. Eng. Syst.*, vol. 12, no. 6, pp. 236–248, Dec. 2019, doi: 10.22266/ijies2019.1231.23.

[6] R. Sarno, F. Sinaga, and K. R. Sungkono, “Anomaly detection in business processes using process mining and fuzzy association rule learning,” *J. Big Data*, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-019-0277-1.

[7] S. K. L. M. Vanden Broucke, J. De Weerdt, J. Vanthienen, and B. Baesens, “Determining process model precision and generalization with weighted artificial negative events,” *IEEE Trans. Knowl. Data Eng.*, 2014, doi: 10.1109/TKDE.2013.130.

[8] A. F. Syring, N. Tax, and W. M. P. van der Aalst, “Evaluating Conformance Measures in Process Mining Using Conformance Propositions,” 2019, doi: 10.1007/978-3-662-60651-3_8.

[9] W. Van Der Aalst, “Process mining: Overview and opportunities,” *ACM Transactions on Management Information Systems*. 2012, doi: 10.1145/2229156.2229157.

[10] T. Gneiting and P. Vogel, “Receiver Operating Characteristic (ROC) curves,” *arXiv*. 2018, doi: 10.1007/978-1-4419-9863-7_242.

[11] T. Fawcett, “ROC graphs: Notes and practical considerations for researchers,” *Mach. Learn.*, 2004.

[12] J. Davis and M. Goadrich, “The relationship between precision-recall and ROC curves,” 2006, doi: 10.1145/1143844.1143874.

[13] K. Hajian-Tilaki, “Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation,” *Caspian Journal of Internal Medicine*. 2013.

[14] D. M. M. Schunselaar, T. Slaats, F. M. Maggi, H. A. Reijers, and W. M. P. van der Aalst, “Mining hybrid business process models: A quest for better precision,” 2018, doi: 10.1007/978-3-319-93931-5_14.

[15] A. Bogarín, R. Cerezo, and C. Romero, “Discovering learning processes using inductive miner: A case study with learning management systems (LMSs),” *Psicothema*, 2018, doi: 10.7334/psicothema2018.116.

[16] W. Pulsanong, P. Porouhan, S. Tumswadi, and W. Premchaiswadi, “Using inductive miner to find the most optimized path of workflow process,” 2018, doi: 10.1109/ICTKE.2017.8259633.