Analysis of Loan Process Using Various Process Mining Techniques: The BPIC 2017

Dohyeon Ryu[1], Jitaek Lim[1], Dawoon Jeong[1], and Minseok Song[1]

[1] POSTECH, Kyungsangbuk-do Pohang 37673, Republic of Korea

Abstract. As process mining has been actively studying, various process mining techniques are emerging. Accordingly, we analyzed the loan process from a bank in Netherlands, focusing on three challenges in BPIC 2017 and further analyses using diverse process mining techniques. The process consists of 561,671 events and 31,509 cases. Before analyzing the process, understanding data and process is conducted in advance. Then, we proceed to 3 questions and further analyses. In order to answer three challenges provided and for further analyses, many analysis techniques such as dotted chart analysis, decision point analysis, user analysis using trace clustering techniques and self organizing map were used. Moreover, various tools were used such as ProM, Disco, excel, python, and R. The results can be utilized in three viewpoints: understanding the process, identifying problems of process, and improving processes. Therefore, the insights from the analyses can provide process managers an opportunity to create more values for their applicants. Also, procedures and techniques used in this paper can be as a reference for other similar processes.

Keywords: Process Mining, Data Analysis, Loan Application

1 Introduction

According to the applicability of IT and devices, it is possible to collect log data in real time. Log data means the data that includes cases, time, activities, and resources of a process. This data can be used in many purposes. For example, it can be used for process identification, process improvement, new process development, etc. Therefore, many industries have been trying to collect and analyze log data to create more values for their customers. Also, many tools to analyze log data have been developed.

Process mining has emerged as a way to analyze the behavior of an organization by extracting knowledge from process-related data, and offering techniques to discover, monitor and enhance real processes [1]. Process mining makes it possible that process managers can understand current execution based on observations recorded in the event log. Process mining consists of three parts (Fig. 1), discover a process model, check the

conformance of a process model, and enhance a process model with performance information or animations. Discovering a process model means finding a process model from the observations of customers, machines, etc. Based on the developed model, conformance checking is conducted. Simply, it means checking whether the model represents the actual process. It is important to develop a proper model based on a purpose of analysis. Enhancing a process model means improving performance of processes.

In this project, data collected to analyze includes processes about loan application. In order to accept a loan, customers and workers of bank have several activities to apply loans and evaluate the applications. The data includes who apply a loan and which activities should be executed to accept or reject the application. Also, the data includes several personal information. The data is more deeply explained in section 2.

Several techniques were conducted to cover the two parts of process mining, discovering a model and checking a conformance. In order to analyze event logs, we used several tools such as python, ProM, Excel, and R. Mostly, we use ProM that provides many plug-ins for beginners. When we need to customize plug-ins provided in ProM, we used other tools. By using the tools, we suggest analysis results and interpretations for questions as follow:

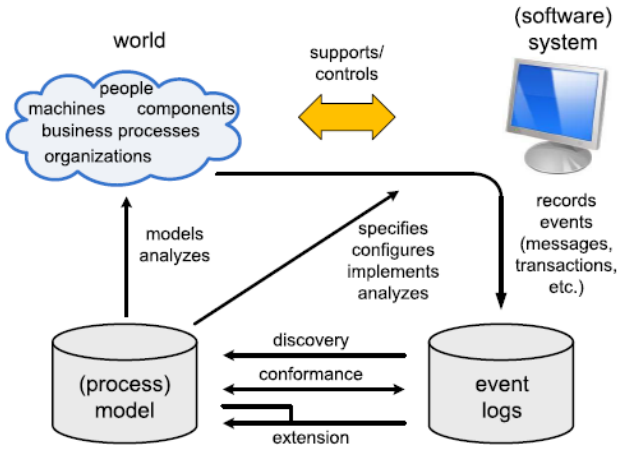

Fig. 1. Three types of process mining: (1) Discovery, (2) Conformance, and (3) Extension (from [2])

1. What are the throughput times per part of the process, in particular the difference between the time spent in the company's systems waiting for processing by a user and the time spent waiting on input from the applicant as this is currently unclear?
2. What is the influence on the frequency of incompleteness to the final outcome. The hypothesis here is that if applicants are confronted with more requests for completion, they are more likely to not accept the final offer?

3. How many customers ask for more than one offer (where it matters if these offers are asked for in a single conversation or in multiple conversations)? How does the conversion compare between applicants for whom a single offer is made and applicants for whom multiple offers are made?
4. Any other interesting trends, dependencies etc.

This paper is structured as follows. In section 2, we first understand data we analyze and overall processes by using dotted chart analysis. Next, the analysis results of questions mentioned above are introduced in section 3, 4, and 5 respectively. From section 6 to 9, additional analyses are introduced to provide some useful insights. After that, the results we conducted are summarized in section 10. Finally, some discussed issues and future research works are mentioned in section 11.

2 Understanding Overall Process

2.1 Understanding Data

The process represents loan process of the bank in Netherlands. A customer makes an application to loan then the bank checks the application and makes an offer or offers. After that, the bank and the customer go through series of steps for the loan. The process has 561,671 events and 31,509 cases. This process was captured from 01/01/2016 to 01/02/2017. There are 26 activities that are classified into three activity types: Application(A_), Offer(O_), and Work item(W_). 10 activities related to Application represent states of the application itself. 8 Offer type activities express states of an offer communicated to the customer. 6 Work item activities show states of work items performed by the bank's employees that occur during the approval process.

The process has only one start activity, A_Create Application and 14 end activities. However, there might be incomplete cases in the process so cases whose endpoint are one among A_Pending, O_Refused, and O_Cancelled were selected as complete cases (Table 1). Although A_Canclled and A_Denied could be endpoints as end state of unsuccessful applications, they were ignored because the number of cases with these two endpoints is only 194 (0.6%). Only complete cases are used in process mining and analysis of incomplete cases existss at section 4.

Table 1. Three endpoints in the process

Endpoint	Description	# of cases (%)
A_Pending	End sate of successful applications At this point, the loan is final and the customer is payed if all documents are received and the assessment is positive.	12791 (40.6%)
O_Refused	End statd of unsuccessful offer At this point, the back refused the customer's offer.	3719 (11.8%)
O_Cancelled	End state of unsuccessful offer At this point, the customer or the bank cancelled the customer's offer.	14707 (46.7%)

2.2 Dotted Chart Analysis

Dotted chart analysis shows events as dots over time in graphical way to get helicopter view of the process [3]. In this paper, dotted chart analysis using ProM 6 was conducted to understand overview of the process.

Constant Arrival Rate. When events are spread according to timestamp (X-axis) and case IDs (Y-axis) sorted by start time of case, it can be seen that the process has almost constant arrival rate (Fig. 2). In Fig. 2, the different colors mean each activity.

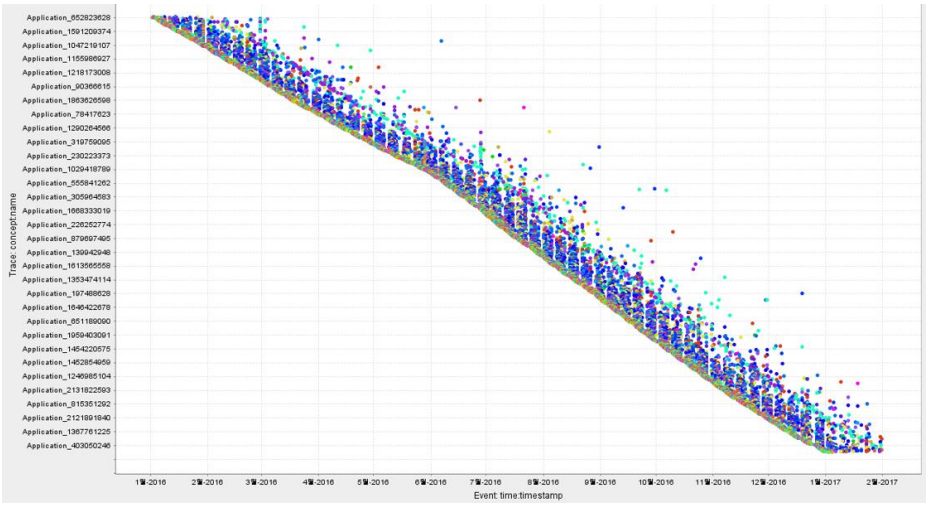

Fig. 2. Result of dotted chart analysis – constant arrival rate

Resource System. Through dotted chart with timestamp (X-axis) and resource (Y-axis) (Fig. 3), we could guess there are two-type of resource. One is automatic system and the other one is not automatic system. This is because User_1 who is assumed to be automatic system works without interruption. On the other hand, other users have

holidays and working hours (Fig. 4, Fig. 5). They rest on Sundays and work about nine hours per day. In Fig. 3, the color represents an activity so we can guess some resources have their roles. This is because some resources (from User_111 to User_137) have similar colored dots. This insight would be the motivation to do section 6.

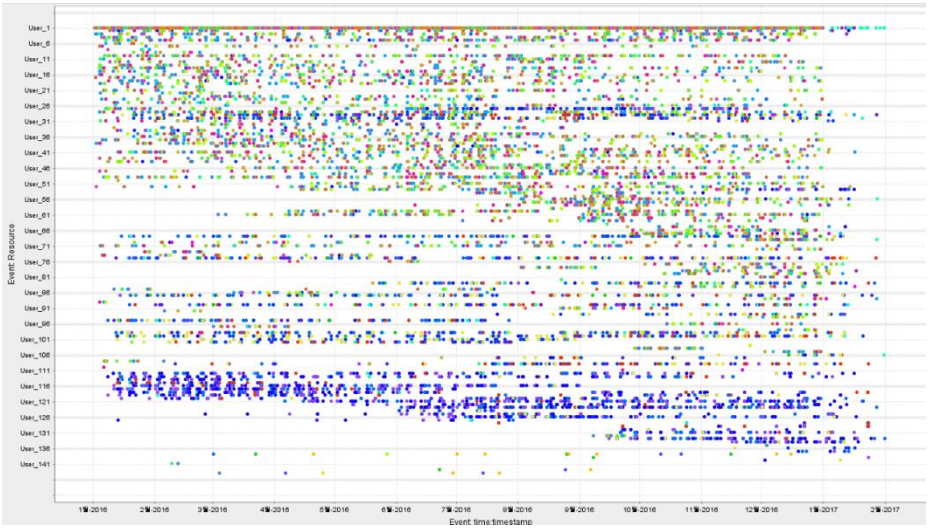

Fig. 3. Result of dotted chart analysis – Resource type

Fig. 4. Result of dotted chart analysis (Expansion of Fig. 3) – Rested on Sundays

Fig. 5. Result of dotted chart analysis (Expansion of Fig. 3) – Working time

Activity types based on resource types. When classification of resources is applied to the result of dotted chart in Fig. 6, we can also make three types of activities like Table 2. In the system, only User_1 is automatic system based on the results of dotted chart analysis of resource system and color shows a resource in Fig. 6. Therefore, activities are classified based on the presence or absence of User_1 which is represented as magenta dot.

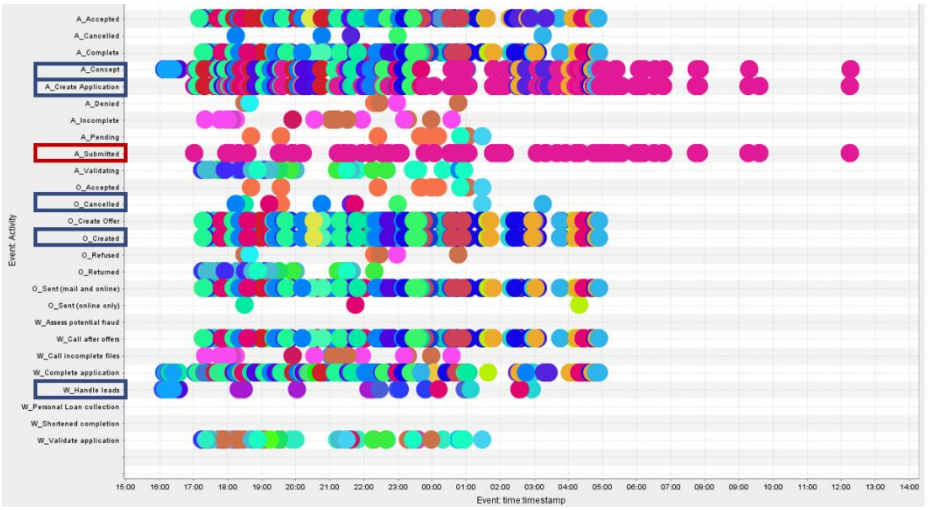

Fig. 6. Result of dotted chart analysis – three activity types based on resource types

Table 2. Three activity types based on resource types

Activity type	Activity name
Activity conducted by only automatic system	A_Submitted
Activity conducted by automatic system and not automatic system	A_Concept, A_Create Application, O_Cancelled, O_Created, W_Handle leads
Activity conducted by only not automatic system	The rest

3 Challenge 1: what are the throughput times per part of the process?

Before answering the question, we needed to define what 'part' means in this question. We assume that 'part' as 'activities' that are executed to accept or reject a loan application. Therefore, the question is paraphrased as 'what are the throughput times per activity of the process?' with our assumption. Typically, throughput time means the total time to finish an activity. Therefore, we should know when the activity starts and finishes. The data recorded includes information about starting and finishing time of each activity. We used this information to calculate throughput time per activity of the process.

Throughput time means the sum of waiting time(WT) and execution time(ET) of activities. However, it is not easy to calculate and define a certain value as waiting time because there are so many prior activities for one next activity. Therefore, we calculated waiting times for all prior activities. As a result, an activity can have many throughput time depending on how many prior activities it has (Fig. 7).

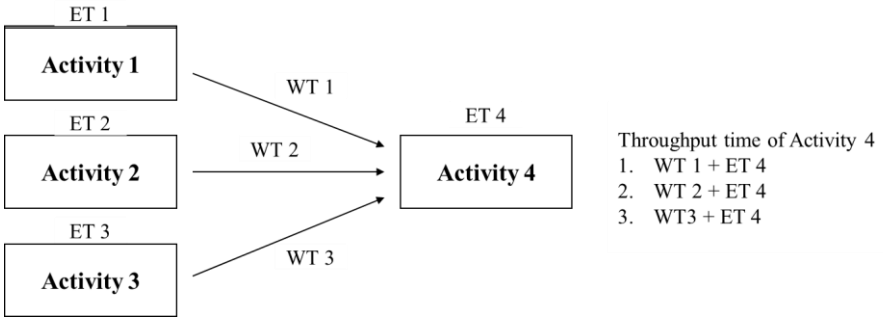

Fig. 7. Throughput time calculation of each activity with waiting times and execution times

By doing this, we also found that there are some activities conducted by automatic systems. If activities are conducted by not automatic system, it will take more than 0

milliseconds. Therefore, we assumed that the activities that take time less than 0 milliseconds of execution time are conducted by automatic systems.

We first found which activity is conducted by automatic system or not automatic system. Then, calculated waiting time and execution time to calculate throughput time. Finally, we drew Fig. 8 as below. As a result, there are 6 activities that are conducted by both automatic system and not automatic system among all 26 activities. There are no activities that are only conducted by not automatic system. The activity that takes the longest (3.1 days) execution time is W_Assess potential fraud. Also, when considering just two activities, prior and next, the longest throughput time is 12 days of W_Assess potential fraud.

Activity	System	Human	Waiting			Execution			Throughput Time (Mean)	Prior Activities
			Max	Min	Mean	Max	Min	Mean		
A_Create Application	O		0 millis	0 millis	0 millis	0 millis	0 millis	0 millis		
A_Submitted	O		25.6 sec	29 millis	339 millis	0 millis	0 millis	0 millis	339 millis	A_Create Application
A_Concept	O		17.4 hrs	18.6 secs	78.2 sec	0 millis	0 millis	0 millis	78.2 sec	A_Submitted
			3.9 secs	1 millis	21 millis				21 millis	A_Create Application
			196 millis	1 millis	5 millis				5 millis	W_Complete application
			4.8 d	8 millis	4 mins				4 mins	W_Handle leads
A_Complete	O		313 millis	19 millis	43 millis	0 millis	0 millis	0 millis	43 millis	O_Sent (mail and online)
			199 millis	19 millis	45 millis				45 millis	O_Sent (online only)
			287 millils	1 millis	3 millis				3 millis	W_Call after offers
A_Accepted	O		3.3 d	4 hrs	37.9 hrs	0 millis	0 millis	0 millis	37.9 hrs	W_Assess potential fraud
			5 d	118 secs	9.8 hrs				9.8 hrs	W_Shortened completion
			30.4 d	18 secs	37.4 hrs				37.4 hrs	W_Complete application
			30.4 d	18.4 secs	23.9 hrs				23.9 hrs	A_Concept
A_Validating	O		44 secs	48 millis	508 millis	0 millis	0 millis	0 millis	508 millis	W_Validate application
			13.1 wkss	1.5 secs	63.9 hrs				63.9 hrs	A_Incomplete
			42 d	61 millis	3.7 d				3.7 d	W_Call incomplete files
			29.7 d	84.1 secs	7.7 d				7.7 d	O_Cancelled
			18.9 d	46.1 hrs	11 d				11 d	W_Shortened completion
			27.8 d	5.7 secs	4 d				4 d	O_Sent (online only)
			30.1 d	10.4 secs	7.7 d				7.7 d	O_Sent (mail and online)
			13.8 d	43.2 mins	5.8 d				5.8 d	O_Created
			30.3 d	2.3 secs	8.7 d				8.7 d	A_Complete
			20.8 d	70 millis	3.2 d				3.2 d	W_Call after offers
			20.1 d	2.9 secs	30.3 hrs				30.3 hrs	O_Returned
			40 secs	32.7 secs	36.3 secs				36.3 secs	A_Validating
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Fig. 8. The table for throughput time of each activity

4 Challenge 2: What is the influence on the frequency of incompleteness to the final outcome?

BPIC 2017 gave us the hypothesis that if applicants are confronted with more requests for completion, they are more likely to not accept the final offer. The frequency analysis of cases ending with A_Pending and O_Cancelled was performed to test this hypothesis. About half of complete cases did not pass A_incomplete. As the number of A_incomplete increased, the number of cases decreased (Fig. 9).

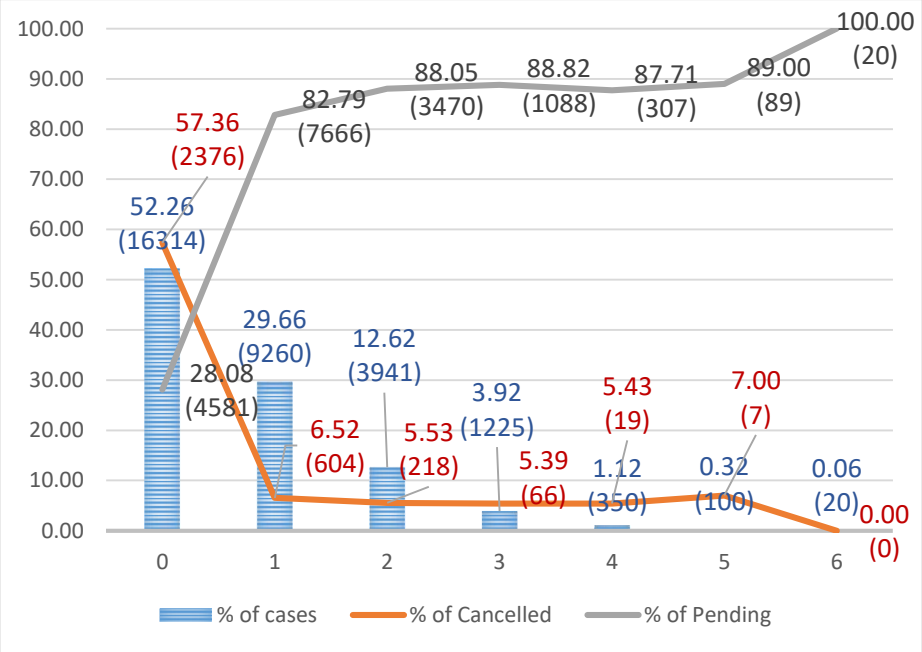

Fig. 9. Changes in the ratio of cases and ending with A_Pending and O_Cancelled

According to Fig. 9, as the number of A_Incomplete increases, A_Pending trend increases and O_Cancelled trend decreases. However, the amount of increase and decrease is not that large. This results can be interpreted as if customers pass more A_Incomplete activity for completion, they are more likely to accept the final offer and end with A_Pending. Therefore, the hypothesis from BPIC 2017 should be rejected.

5 Challenge 3: How many customers ask for more than one offer and how does that relate to conversion

5.1 How many customers ask for more than one offer

This section provides answers to the following question: How many customers ask for more than one offer and how does that relate to conversion? As a first step, we counted the number of applicants who offer more than once. As the number of offers is only in the BPI Challenge 2017 – Offer log, we focused on this dataset. Then, the following steps were conducted:

1. Read the dataset into Python
2. Group 'Case ID' by '(case) Application ID' in BPI Challenge 2017 – Offer log
3. Count the number of set of 'Case ID' by each '(case) Application ID'
4. Remove '(case) Application ID' which has only one type of 'Case ID'
5. Export the data_1

As a result of above procedures, there are 8469 customers who ask for more than once. Fig. 10 shows six customers among 8469 customers as examples.

application	offernum
Application_1000671285	['Offer_620142850', 'Offer_620142850', 'Offer_620142850', 'Offer_888834469', 'Offer_888834469', 'Offer_888834469', 'Offer_888…
Application_1000691650	['Offer_1595985595', 'Offer_1595985595', 'Offer_1595985595', 'Offer_1595985595', 'Offer_1108897743', 'Offer_1108897743', 'Offer_…
Application_1001114274	['Offer_1360642045', 'Offer_1360642045', 'Offer_1360642045', 'Offer_1360642045', 'Offer_173087468', 'Offer_173087468', 'Offer_17…
Application_1002485344	['Offer_285108194', 'Offer_285108194', 'Offer_285108194', 'Offer_285108194', 'Offer_285108194', 'Offer_817645433', 'Offer_817…
Application_1002626536	['Offer_798569591', 'Offer_798569591', 'Offer_798569591', 'Offer_798569591', 'Offer_798569591', 'Offer_2025975087', 'Offer_20…
Application_1002664914	['Offer_592386587', 'Offer_592386587', 'Offer_592386587', 'Offer_110985706', 'Offer_110985706', 'Offer_110985706', 'Offer_110…

Fig. 10. Grouping Offer ID by Application ID

5.2 How does that relate to conversion

Based on the above result, we moved to answer the next question: How does that relate to conversion? Before we do that, we first defined the concept of conversion: conversion in this case means that the application arrives to the end state A_Pending, where the loan is actually paid to the customer. To extract the cases that have the state of A_Pending, we employed process mining analysis in Disco. As activity information is only contained within BPIC 2017, we focused on this dataset when extracting cases which contain the activity, A_Pending. Therefore, the following steps were conducted:

1. Import of BPIC 2017 data into Disco
2. Extraction of the cases which have A_Pending activity using filter of attribute
3. Export the data_2

By following this procedure, we obtained the ID of applicants who have the activity of A_Pending. In order to find the number of applicants who have A_Pending activity and offer more than two offers, we checked whether the '(case) Application ID' in data_1 is in 'Case ID' in data_2 and obtained the matched one. Therefore, there are 5050 applicants (30%) who offer more than two offers among 17227 applicants who arrive to the end state of A_Pending.

In order to know the difference between the applicants who offer one offer and more than two offers, we did T-Test on 5 attributes; '(case) OfferedAmount', '(case) NumberOfTerms', '(case) FirstWithdrawalAmount', '(case) MonthlyCost', and '(case) CreditScore'.

(case) OfferedAmount

	More than two offers	One offer
Mean	18994.54	18700.44
Variance	1.82E+08	1.57E+08
Observations	614	614
Hypothesized Mean Difference	0	
df	1219	
t Stat	0.39604	
P(T<=t) one-tail	0.346072	
T Critical one-tail	1.646105	

(case) NumberOfTerms

	More than two offers	One offer
Mean	85.99811	83.21863
Variance	1.33E+03	1318.607
Observations	614	614
Hypothesized Mean Difference	0	
df	1226	
t Stat	1.338026	
P(T<=t) one-tail	0.090568	
T Critical one-tail	1.646097	

(case) FirstWithdrawalAmount

	More than two offers	One offer
Mean	7796.578	8750.693
Variance	8.38E+07	91031441
Observations	614	614
Hypothesized Mean Difference	0	
df	1224	
t Stat	-1.78786	
P(T<=t) one-tail	0.037023	
T Critical one-tail	1.646099	

(case) MonthlyCost

	More than two offers	One offer
Mean	278.6283	301.9735
Variance	3.12E+04	55026.05
Observations	614	614
Hypothesized Mean Difference	0	
df	1139	
t Stat	-1.9703	
P(T<=t) one-tail	0.024523	
T Critical one-tail	1.646193	

(case) CreditScore

	More than two offers	One offer
Mean	766.9006515	469.2563
Variance	1.08E+05	22526.59
Observations	614	614
Hypothesized Mean Difference	0	
df	859	
t Stat	20.43826223	
P(T<=t) one-tail	2.86E-76	
T Critical one-tail	1.64662944	

Fig. 11. T-test of 5 attributes

As you can see from the results (Fig. 11), there are 3 attributes '((case) FirstWithdrawalAmount', '(case) MonthlyCost', and '(case) CreditScore' which show the significant difference between the group of one offer and that of more than two offers (under $\alpha = 0.05$).

6 Further analysis: Analysis of Incomplete Cases

As mentioned in section 2, the process includes incomplete cases. According to [4], incomplete case is missing either the start or the end of the process. Based on classification of reasons for why a case is incomplete in [4], we divided incomplete cases into two types:

Type 1: Cases not yet finished
Type 2: Cases never ended

Type 1 cases can be ended without any problems if they have more time. Whereas Type 2 cases might have some problems in the process so they cannot end no matter how much time they spend. Time duration from last activity complete time to data extraction time is the criterion that distinguishes between two types. If the time duration exceeds the maximum time duration from the activity which is the last activity in the incomplete case to the next activity, we considered the case would not be over [4]. Otherwise, we thought that case would be finished sometime.

Actually, [4] classified reasons why incomplete case occurs into three ways. Only two of them were utilized. The other one is data extraction method which has retrieved only events in a certain time frame. The reason why data extraction method was excluded is that it is difficult to distinguish cases with data extraction method problem and cases not yet finished unless checking start time of cases. Therefore, start time of incomplete cases should be considered later to find more specific reason.

6.1 Type 1: Cases not yet finished

Cases not yet finished were total 97 and ended in one of seven activities (Table 3).

Table 3. Number of type 1 cases due to last activity

Last activity	Last activity	Number of cases	Max. time in incomplete cases
A_Incomplete	30	89.1 days	13.2 weeks
A_Complete	30	30.2 days	32.6 days
O_Sent (mail and online)	19	30.2 days	17 weeks
O_Sent (online only)	12	28.1 days	87.9 days
A_Validating	3	6.4 hours	49 days
O_Returned	2	8 days	13.2 weeks
W_Call after offers	1	22.3 days	73.9 days

Especially, A_Complete duration in incomplete cases (median duration: 29.2 days, mean duration: 27.9 days, max. duration 30.2 days, min. duration: 14 days) is longer than complete cases (median duration: 7.1 days, mean duration: 8.8 days, max. duration: 30.3 days, min. duration: 1.6 secs). Incomplete cases may have factors causing time delay comparing to complete cases.

The remaining time of the incomplete case can be predicted using FSM (Finite State Machine) miner. We tried to calculate the remaining time of cases ending with O_Sent (online only) based on FSM analyzer method [5]. There are 8 cases which have a pattern in Fig. 12. The remaining time of each case is as follows;

1. 7 days 14 hours
2. 10 days
3. 8 days 21 hours
4. 6 days 19 hours
5. 10 days 19 hours
6. 3 days 21 hours
7. 15 days 3 hours
8. 8 days 14 hours

	Activity
1	A_Create Application
2	A_Submitted
3	W_Handle leads
4	W_Complete application
5	A_Concept
6	A_Accepted
7	O_Create Offer
8	O_Created
9	O_Sent (mail and online)
10	W_Call after offers
11	A_Complete
12	O_Create Offer
13	O_Created
14	O_Sent (online only)

Fig. 12. The pattern with O_Sent (online only)

In order to predict the remaining time of incomplete case with O_Sent (online only), the average the 8 remaining times and its value is 215 hours (about 9 days). Likewise, with respect to other activities, we apply this FSM miner and can predict remaining time of incomplete cases.

6.2 Type 2: Cases never ended

The process has only two cases of type 2. One ended with W_Shortened completion and the other one finished with W_Personal Loan collection (Table 4).

Table 4. Number of type 2 cases due to last activity

Last activity	Number of cases	Time in incomplete cases	Max. time in complete cases
W_Shortend completion	1	29.3 weeks	30.9 days
W_Personal Loan collection	1	30.3 weeks	75.2 mins

The customer who has a certain profile that defines a lower credit risk passes W_Shortened completion. Therefore, the case ending with W_Shortend completion might have some problems such as missing the application or documents.

The case ending with W_Personal Loan collection has unusual points outside of duration time. W_Personal Laon collection happens before A_Pending in the process of complete cases. However, in the incomplete case, W_Personal Laon collection was followed by A_Pending. We thought it happened because the bank failed to process A_Pending well or missed W_Personal Loan collection in application validating steps.

7 Further analysis: Decision point analysis

7.1 Exploring decision point

Within this section, we give answers to this question: Are there any decision rules among decision points? As a first step, we found the decision points. The following procedures were conducted:

1. Import of the BPIC 2017 to Disco
2. Set “activities” slider to 100% & “Paths” slider to 0%

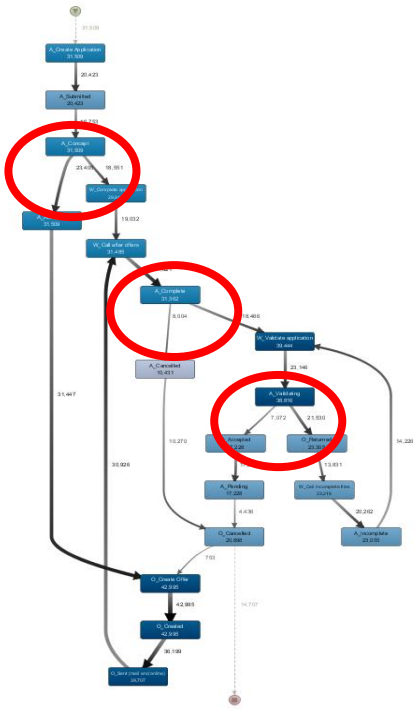

Fig. 13. The process map whose activities set is 100% and paths set is 0%

As you can see from map, there are 3 decision points (Fig. 13).

1. A_Concept → (A_Accepted, W_complete application)
2. A_Complete → (A_Cancelled, W_Validate application)
3. A_Validating → (O_Accpeted, O_Returned)

We conducted decision point analysis to 3 decision points above, but we could not obtain good results of first and third decision points. Therefore, the result of second decision point was only obtained.

Before decision point analysis, we needed to know the type of split (AND-split, OR-split, XOR-split). In Disco, we applied the filer of directly followed between A_Complete and A_Cancelled (Fig. 14).

Activity	Resource	Date
A_Create Application	User_1	01.01.2016
A_Submitted	User_1	01.01.2016
A_Concept	User_1	01.01.2016
W_Complete application	User_19	02.01.2016
A_Accepted	User_19	02.01.2016
O_Create Offer	User_19	02.01.2016
O_Created	User_19	02.01.2016
O_Sent (mail and online)	User_19	02.01.2016
W_Call after offers	User_19	02.01.2016
A_Complete	User_19	02.01.2016
A_Cancelled	User_1	02.02.2016
O_Cancelled	User_1	02.02.2016

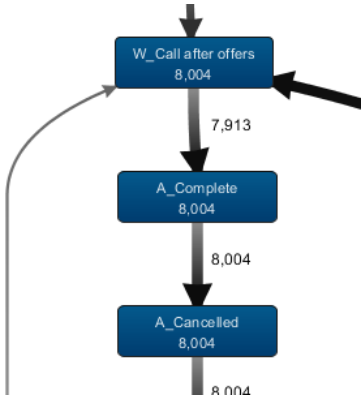

Fig. 14. The relationship between A_Complete and A_Cancelled.

Likewise, we applied the filter of directly followed between A_Complete and W_Validate application (Fig.15).

Activity	Resource	Date
A_Create Application	User_1	01.01.2016
A_Submitted	User_1	01.01.2016
A_Concept	User_1	01.01.2016
W_Complete application	User_17	02.01.2016
A_Accepted	User_52	02.01.2016
O_Create Offer	User_52	02.01.2016
O_Created	User_52	02.01.2016
O_Sent (mail and online)	User_52	02.01.2016
W_Call after offers	User_52	02.01.2016
A_Complete	User_52	02.01.2016
W_Validate application	User_117	13.01.2016
A_Validating	User_117	13.01.2016

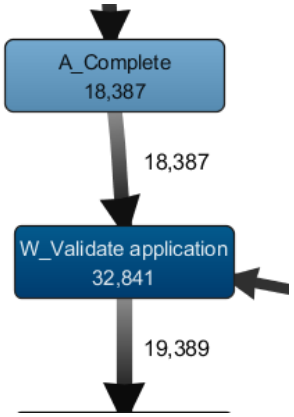

Fig. 15. The relationship between A_Complete and W_Validate application

We saw all the variants of both filtered ones, but could not find the evident of AND-split and OR-split. Therefore, we could conclude that this decision point is XOR-split.

There are 6 variables which could be used in the features of decision tree. We drew all possible decision tree using 1 feature, 2 features, 3 features, 4 features, 5 features and lastly all features.

Among them, we could obtain the best decision tree whose features are '(case) FirstWithdrawalAmount, '(case) CreditScore', and '(case) MonthlyCost'.

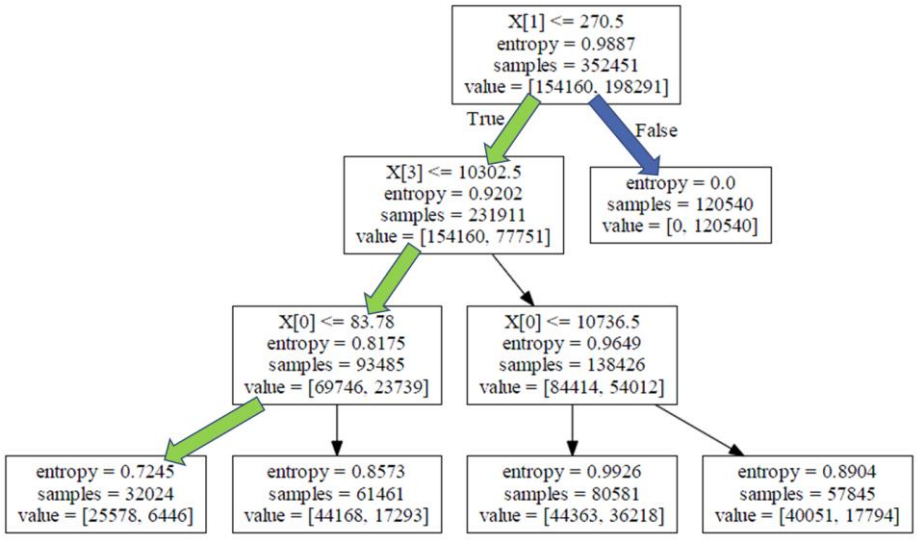

Fig. 16. A decision tree with 3 attributes; '(case) FirstWithdrawalAmount, '(case) CreditScore', and '(case) MonthlyCost'.

In this decision tree which is shown in Fig. 16, we could obtain two conclusions

1. If CreditScore > 270.5, then it goes to W_Validate application
2. If CreditScore <= 270.5 and OfferedAmount <=10302.5 and FirstWithdrawalAmount <=83.78, then it goes to A_Cancelled.

8 Further analysis: User analysis

8.1 Exploring the number of resources per activity

Within this section, we give answers to this question: Are there any teams or functional structures of users. Before the analysis, we plotted the graph of how many resources are involved in each activity (Fig. 17).

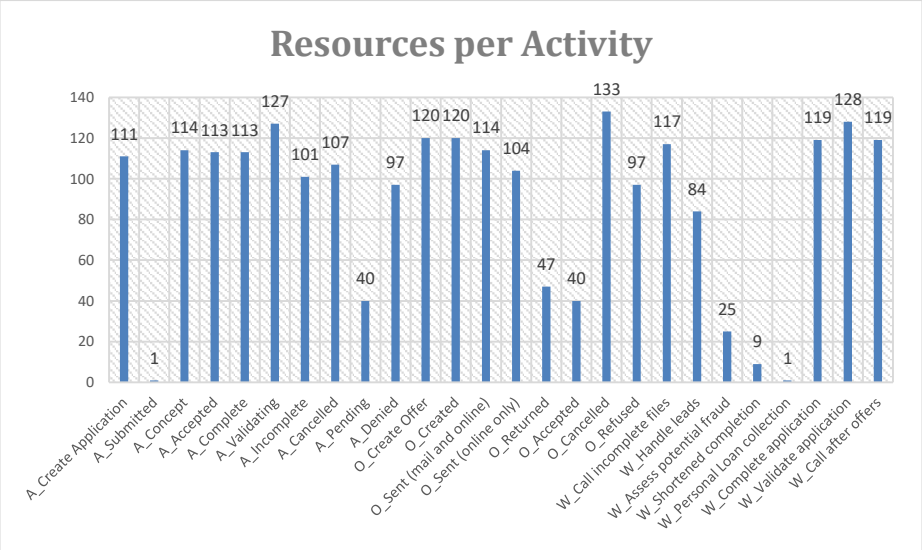

Fig. 17. The number of resources involved in each activity.

There are two activities A_Submitted and W_Personal Loan collection which are done only by one resource. Except for them, activities left are done by about 110 resources. In order to find the functional structures, we used Hierarchical clustering algorithm in R. The first thing is to make matrix. This matrix shows the "profile" of a resource based on whether each resource conducts certain activities or not [6] The matrix is defined as follows

Definition 1. (Δ) *Let* $\mathrm{T} = \{t_1, t_2, \ldots t_m\}$ *be a set of tasks,* $P = \{p_1, p_2, \ldots p_n\}$ *be set of performers in a business process. A function* π_t *indicates the task of a given event* c_i*, while a function* π_p *indicates the performer of a given event. Then*

$$p_1 \Delta t_1 = \begin{cases} 1 & if\ \pi_t(c_i) = t_1\ and\ \pi_p(c_i) = p_1 \\ 0 & otherwise \end{cases}$$

The metric based on joint activities or the profile of performers can be represented in a matrix form as shown in Table 5. The value in the matrix indicates whether each performer conducts a specific task or not. For example, the performer P1 conducts Task A and Task B according to Table 5.

Table 5. An example of a matrix.

	Task A	Task B	Task C	Task D
P1	1	1	0	0
P2	1	1	0	0
P3	0	1	1	0

With this matrix, the following procedures were conducted:

1. Import the matrix to R
2. Apply Hierarchical clustering with 5 clusters
3. Draw the Dendrogram

Then, we obtained the Dendrogram in Fig. 18.

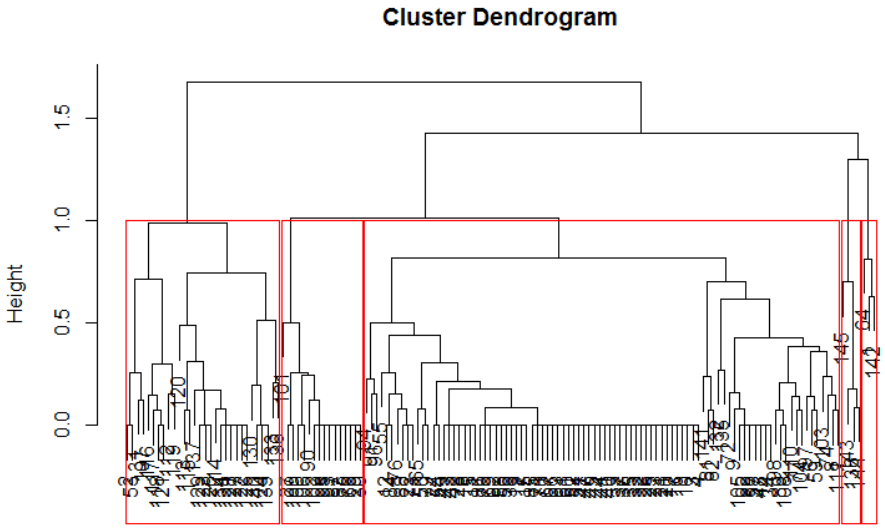

Fig. 18. Dendrogram.

1. Cluster 1: 1, 64, 142
2. Cluster 2: 2, 53, 107, 112~131, 133, 134, 136, 137, 139, 140
3. Cluster 3: the rest
4. Cluster 4: 27, 29, 30, 68, 75, 83, 87, 90, 93, 95, 99, 100, 101, 102, 106, 109
5. Cluster 5: 138, 143, 144, 145

Resources are involved in the above 5 clusters. Then, we checked these clusters with resource type dotted chart (Fig. 19) mentioned in section 2.2.

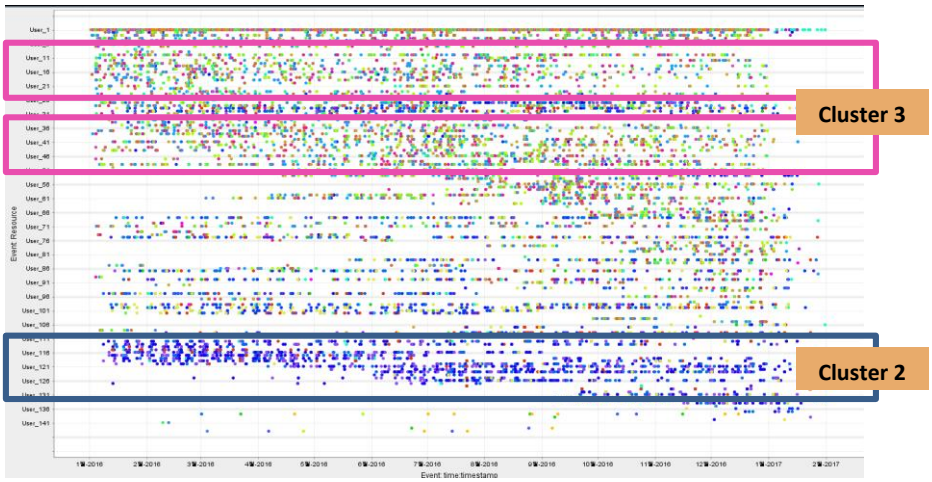

Fig. 19. Result of dotted chart analysis – Resource type

The red parts are matched to cluster 3 and blue part is matched to cluster 2. Therefore, the result of this analysis can somewhat show the teams or functional structures of given resources.

Organizational miner helps a process manager find groups or teams of workers in the process. We expected to find some groups of all 'User'. As decision analysis, we randomly extracted 15,000 samples and used Organizational Miner plug-in in ProM. Among many methods, we selected 'Self Organizing Map Mining'. The options and results we drew exist in Fig. 20. With the result, we cannot find and interpret any meaningful insights. Therefore, we conducted 'Default Mining' to develop an organizational model (Fig. 21). However, the result looks very complex and hard to understand clear teams or groups among 'Users'.

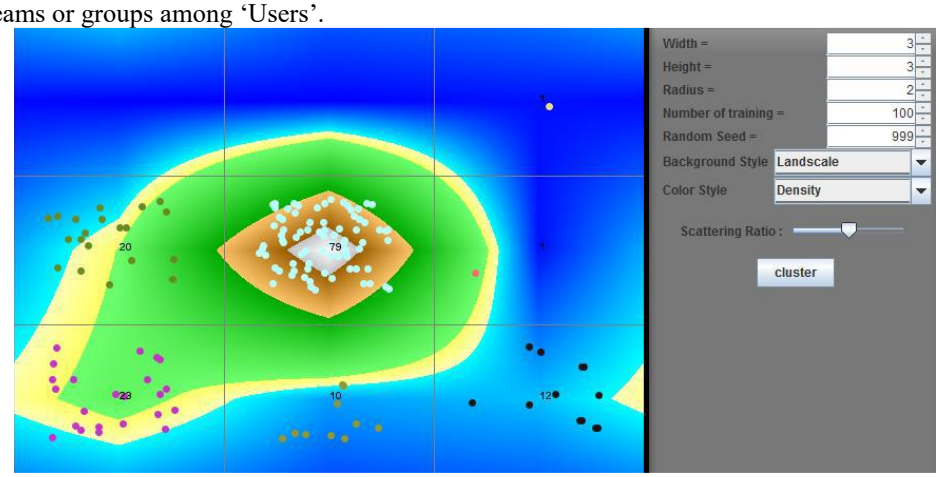

Fig. 20. The result of SOM in Organizational Miner

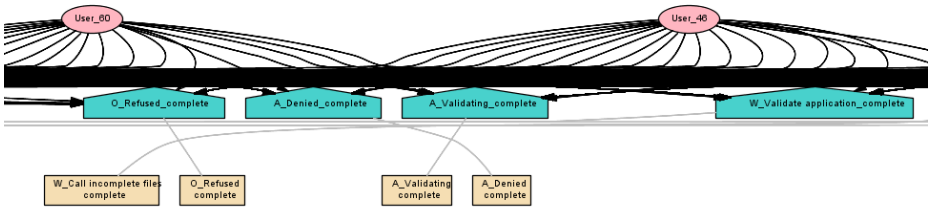

Fig. 21. The result of Default Mining in Organizational Miner

The results of SOM and default mining visualize many users in the process. There are more than 100 users. Also, considering the results, we can know that there are no clear groups or teams that work similar jobs or together. Actually, when we see users manually, we cannot find any rule or groups. It is natural that the results are not clear and look so complex.

The trace clustering technique can be used to group similar cases into homogeneous subsets (clusters) according to their log traces [7]. Making clusters can help understand who works similar jobs and how to manage them. With these reasons, we conducted clustering analysis by using clustering analysis plug-in in ProM. We used K-Means clustering to make 3 clusters of 'User' with the 15,000 samples. Then, we extracted 3 clusters in Fig. 22. In order to verify the result, fitness values and distance values were checked (Fig. 23). In Fig. 23, we can see that the average value of fitness is 0.6526, and it is enough to verify whether users are properly divided into 3 clusters. Also, when we see the distance values, intra distances of each cluster is shorter than the distance between centroids of clusters. It means that clusters are not overlapped each other.

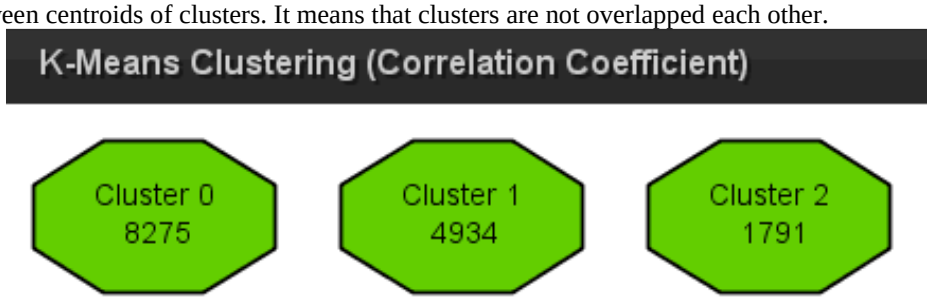

Fig. 22. 3 clusters and number of User assigned to each cluster

Cluster	size	intra distance (avg)
Cluster 0	8275	0.14469413683631152
Cluster 1	4934	0.08647975982765531
Cluster 2	1791	0.12299511836005209

average = 0.11805633834133962

Cluster	size	fitness
Cluster 0	8275	0.6587063670158386
Cluster 1	4934	0.642078697681427
Cluster 2	1791	0.6537781357765198

average = 0.6526485415061315

Distance between centroids

	Cluster 0	Cluster 1	Cluster 2
Cluster 0	0.0	0.20757938815307514	0.1807970910821216
Cluster 1	0.20757938815307514	0.0	0.18326095388823263
Cluster 2	0.1807970910821216	0.18326095388823263	0.0

average = 0.19054581104114313

Fig. 23. The result of fitness values and distance values

With the result of K-Means clustering, we conducted next analysis to find and understand which characteristics each cluster has. We decided to draw process paths of each cluster to see which activities Users of each cluster conduct. First, we drew a process path of cluster 0 (Fig. 24). We found that users assigned to cluster 0 mainly work for a process finished with the activities, O_Accepted and A_Pending. Second, we drew a process path of cluster 1 (Fig. 25). We found that users assigned to cluster 1 mainly work for a process finished with the activities, A_Cancelled and O_Cancelled. Finally, a process path of cluster 2 was drawn (Fig. 26). We found that users assigned to cluster 2 mainly work for a process finished with the activities, A_Denied and O_Refused.

As a result, users of cluster 0 mainly work activities related to loan acceptance and providing customers a loan. However, users of cluster 1 mainly work activities related to applications cancelled by customers. Similarly, users of cluster 2 mainly work activities that users reject applications with any reasons.

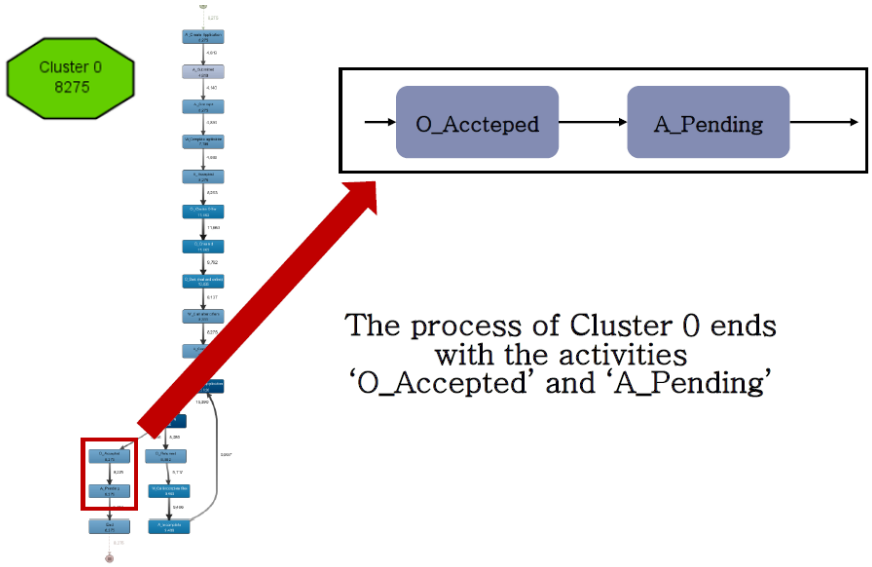

Fig. 24. Process path of cluster 0 and end activities

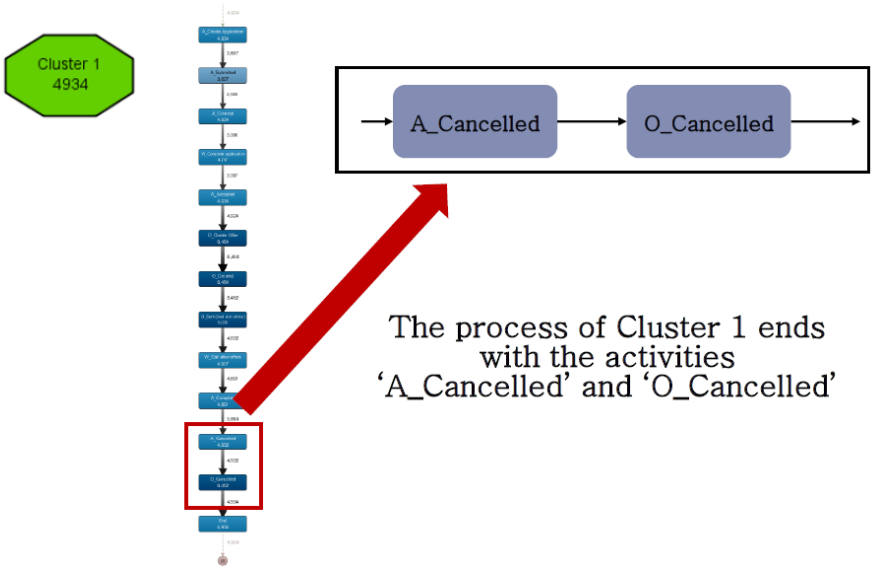

Fig. 25. Process path of cluster 1 and end activities

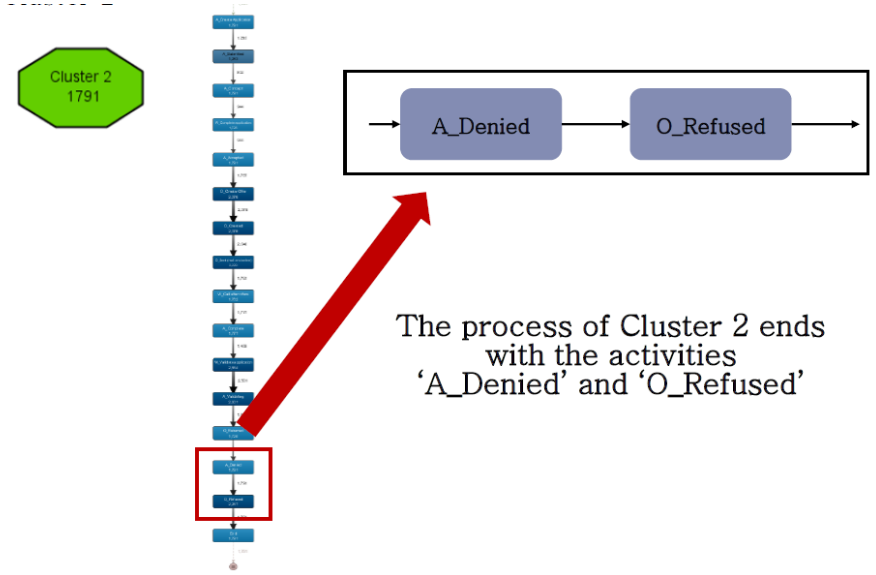

Fig. 26. Process path of cluster 2 and end activities

9 Further Analysis: Analysis of W_Shortened completion

There are cases which go through the activity, W_Shortened completion. Customers who have a certain profile that defines a lower credit risk go through the activity. We decided to analyze and compare process paths between cases which go through W_Shortened completions or not.

First, we drew a process model of cases which do not go through W_Shortened completion (Fig. 27). We checked performance of each activity and found that the activity, W_Assess potential fraud takes much time and acts like a bottleneck in the process.

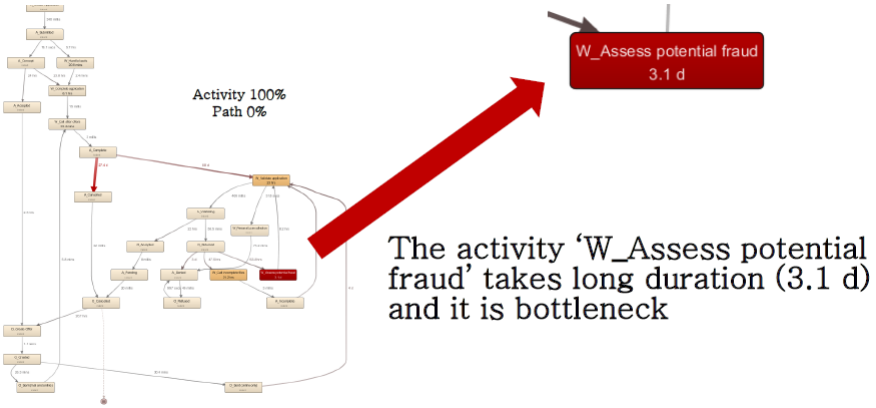

Fig. 27. Process model that contains cases which go through W_Assess potential fraud

When we drew a process model of cases which go through W_Shortened completion (Fig. 27), we found that the cases do not go through the activity, W_Assess potential fraud which takes the longest time in Fig. 28. Finally, we verified whether the process that contains cases going through W_Shortened completion takes less than the process that contains cases not going through W_Shortened completion. Finally, we found that the process that contains cases going through W_Shortened completion takes less (Fig. 29). The process that contains W_Shortened completion activity takes less time in both median case duration and mean case duration. As a result, we can assume that applicants who go through the activity, W_Shortened completion take less time to obtain or reject a loan because they are known as having lower credit risk by banks.

Additionally, we tried to find some characteristics of applicants who go through the activity, W_Shortened completion. So, we found some differences between cases. In Table 6, we can see that applicants who are not defined to have lower credit risk have high risk loan goal such as business goal, debt restructuring, remaining debt home, and tax payment. On the contrary, applicants who are defined to have lower credit risk and go through the activity, W_Shortened completion have comparatively less risk loan goals such as boat, car, motorcycle, home improvement and so on. Also, when we compared requested amount and monthly cost, those of applicants who have lower credit risk have less values in maximum and average values.

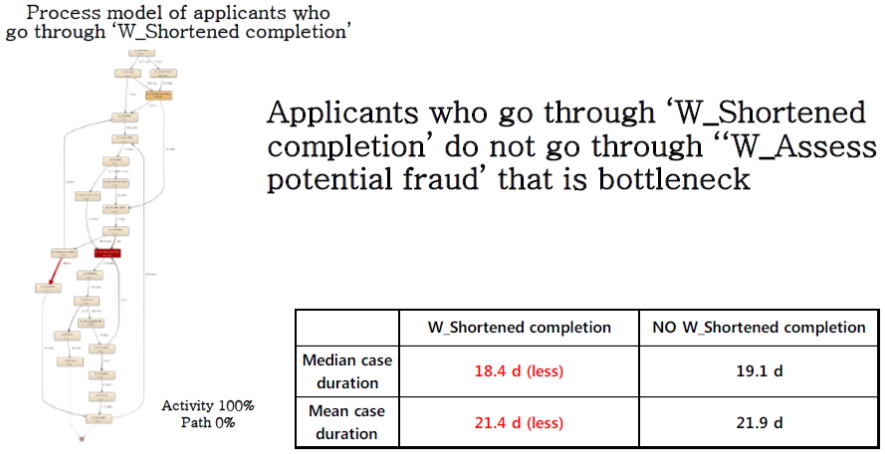

Fig. 28. Process model that contains cases which do not go through W_Assess potential fraud

Table 6. Throughput time of each activity

	W_Shortened completion	No W_Shortened completion
Median case duration	18.4 days (less)	19.1 days
Mean case duration	21.4 days (less)	21.9 days

Considering above results, we can argue that applicants who have less risk of loan goals are evaluated to have lower credit risk. Also, these applicants have a probability

to go through the activity, W_Shortened completion and take less time than others. Process managers may need to make some standards like our assumptions, so that they save their time and cost to verify whether applications are qualified to approve a loan application or not.

	NO W_Shortened completion	W_Shortened completion	Difference
Loan Goal	* Boat * Business goal * Car * Caravan / Camper * Debt Restructuring * Existing Loan Takeover * Extra Spending Limit * Home Improvement * Motorcycle * Not Specified * Other, see explanation * Remaining debt home * Tax Payments * Unknown	* Boat * Car * Caravan / Camper * Existing Loan Takeover * Extra Spending Limit * Home Improvement * Motorcycle * Not Specified * Other, see explanation	* Business goal * Debt Restructuring * Remaining Debt Home * Tax Payment * Unknown
Requested Amount	* Min : 0 * Max : 450,000 * Avg : 16564.40	* Min : 5,000 * Max : 75,000 (less) * Avg : 15162.74 (less)	* Avg : 1401.66
Monthly Cost	* Min : 43.05 * Max : 6673.83 * Avg : 281.40	* Min : 70.5 * Max : 753.49 (less) * Avg : 236.00 (less)	* Avg : 45.4

Fig. 29. The table to compare characteristics between applicants who go through W_Shortened completion or not

10 Conclusion

This paper proposed an approach to understand the overall processes, discover the problems and obtain some insights by analyzing event logs. First, the whole processes which contain only complete cases could be divided into parts and each part follows certain path. Therefore, we categorized the processes into 3 parts using Hierarchical clustering.

Based on the understanding of data, we found some problems of this process. There are some cases never ended due to missing the documents or proceeding with the work in wrong order. In addition, there are some bottlenecks which took long time to be finished.

By using many process mining techniques with various tools such as ProM, Disco, Python, R and Excel, we obtained some insights for improvements to this loan process. Firstly, as the number of A_incomplete increased, then the number of cases decreased, A_Pending trend increased and O_Cancelled trend decreased. Also, if the number of offer is one and that of incomplete is more than once, the probability of finishing the cases with A_Pending is high. With the decision point analysis, there are two insights. If credit score is above 270.5, then it is likely to go path from A_Complete to W_Validate application. Next one is that if credit score is below 270.5, offered amount is below

10302.5 and first withdrawal amount is below 83.78, it is likely to go path from A_Complete to A_Cancelled. Lastly, we could predict the remaining time of incomplete cases based on complete cases with same pattern of path.

11 Conclusion Remarks

The all results of analysis can be categorized into three viewpoints: understanding the process, identifying problems of process, and improving processes. First of all, we conducted dotted chart analysis and clustering analysis for understanding the process. Based on the result of dotted chart analysis, we found when users have holidays and who work constantly with any activities. Also, we found groups or teams in which users work similar activities. These understanding about the loan process help process managers find process flows and define a standard process. Second, we calculated throughput times of each activity and found some cases never ended with some problems. This information can be used to verify whether the process is going well or not. Also, throughput times can be used to find any activities that cause bottleneck in the process. Process managers should find the causes why cases never ended happen in the process. Finally, we found some rules of process that have an opportunity to improve and make the process more efficient. There is a rule related to the number of offer and incomplete. As an offer for loan happen only once and an incomplete condition happens more than once, the probability that applicants obtain the loan is high. Also, we verified that we can calculate the estimated time when the incomplete cases are finished based on the time of other finished activities. In addition, we know that credit score and W_Shortened completion activity can be also used as rules. Process managers can use these rules to improve the process. For example, they can make a small team only for the applicants who are qualified and go through W_Shortened completion and assign other users for teams that need more users. Also, they can save their cost and time to judge acceptance or rejection based on the rules extracted.

Although the results of analysis can be utilized in many purposes as mentioned above, there are some limitations and future research works. First of all, because of the limitation of hardware for analysis, we used 15,000 samples randomly extracted. This may cause difficulties to interpret analyzed results and find some insights. Second, more information of applicants is needed. There are some features of applicants such as credit score, monthly costs, etc. However, when we drew some results based on the analysis, we found that there are not many features that have high correlation with other feature or activities. Therefore, if there are proper and more features of applicants, the analyzed results will have more meaningful insights. Finally, when we conducted analysis related to users, we felt difficult to draw any insights. The information of users is not enough to find any groups or relationships of each other. If more information such as departments and person or system is provided, the more useful interpretation and insights can be extracted.

References

1. Vázquez-Barreiros, Borja, et al. "Process Mining in IT Service Manage-ment: A Case Study." Proceedings of the 2016 International Workshop on Algorithms & Theories for the Analysis of Event Data, ATAED, pp. 16-30, Tourn, Poland (2016).
2. Mans, R. S., van der Aalst, W. M., Vanwersch, R. J., and Moleman, A. J.: “Process mining in healthcare: Data challenges when answering frequent-ly posed questions.” Process Support and Knowledge Representation in Health Care, 140-153 (2013).
3. Song, M., van der Aalast, W. M.: Supporting process mining by showing events at a glace. In: Proceedings of 17th Annual Workshop on Information Technologies and Systems (WITS), pp. 139-145, Montereal, Canada (2007).
4. Fluxicon, How To Deal With Incomplete Cases in Process Mining, http://coda.fluxicon.com/assets/downloads/Articles/PMNews/Incomplete-Cases-Analysis-Process-Mining.pdf.
5. Van der Aalast, W. M., Schonenberg, M. H., and Song, M. Time prediction based on process mining. Information Systems, 36(2), 450-475 (2011).
6. Van der Aalst, W.M.P., Reijers, H., and Song, M. Discovering social networks from event logs. Computer Supported Cooperative Work, 14, 549–593 (2005)
7. Song, M., Günther, C. W., and Aalst, W. M. Trace clustering in process mining. Business Process Management Workshops, 109-120 (2009).