Analysis of Loan Process through Process Mining

Hyeji Jang[1], Yoonseon Jeong[1] and Wongil Kim[1]

[1] POSTECH, 77 Cheongam-Ro. Nam-Gu. Pohang. Gyeongbuk. Korea 37673
[wdfokj, seon1474, sheeppower]@postech.ac.kr

Abstract. A real life event log of the loan and overdraft approvals process from a bank in Netherland is analyzed using process mining and other process mining techniques. The event log consists of 561,617 events and 31,509 cases. In this paper, we first identify some characteristics of event data using dotted chart analysis. After that, we discover the process model using process discovery algorithm and process mining tools like Disco and ProM. Finally, we carry out performance analysis that which one is the bottleneck of this process. Using the result of analysis, we answer the questions from BPI challenge. It is related to (a) what are the throughput times of the process, (b) what is the influence on the frequency of incompleteness to the outcome, (c) how many customers ask for more than one offer. By answering these questions, we present a process model and give insight of the bank loan process.

Keywords: BPIC 2017, process mining, data analysis, loan application

1 Introduction

In today's business supporting information systems generates more and more data, so that the need for analysis of business is increasing. Process mining is a useful technique for analyzing business using event log which contains behaviors. Process means collection of related events, activities and decisions, that involve a number of actors and resources, and that collectively lead to an outcome that is of value to an organization or its customers. Process mining is a process management technique that allows for the analysis of business processes based on event logs. During process mining, specialized data-mining algorithms are applied to event log datasets in order to identify trends, patterns and details contained in event logs recorded by an information system. Process mining aims to improve process efficiency and understanding of processes.

In this paper we report our results from the analysis of a process log provided for the Business Process Intelligence Challenge 2017 (BPI Challenge 2017). The BPI Challenge is an annual international competition in which analyze real-life event log using process mining-related analysis. BPI 2017 data was provided by Dutch Financial Institute which is the same company as BPI 2012. However the company switched systems and the dataset is richer than before. The main difference is that they now support multiple offers for a single application. The event

log contains all applications field in 2016, and their subsequent handling up to Feb 2, 2017.

Each record in the dataset described a single step taken by an applicant or user in the process. We tried to draw the analysis results for 3 main questions which the BPI Challenge 2017 raised which company is particularly interested in answers.

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.

In this year, for the first time, BPI divides participants as three categories, namely students, academics and professionals. We apply the student category which targets Bachelor, Master and PhD students or student teams. The process mining is a mining methodology that identify

2 Used tool

Here, we introduce some tools to analysis our event data. They helped visualization and manipulation for event data.

2.1 DISCO

Disco makes it easy and quick to get started with process mining. It can visualize process maps, process map animation, and powerful log filters in an intuitive way. Disco is fully compatible with the popular academic process mining toolkit ProM due to its support of the event log standards MXML (ProM 5 and ProM 6) and XES (ProM 6). Users can export event logs from Disco and further analyze them with ProM for more advanced courses.

ProM is an extensible framework that supports a wide variety of process mining techniques in the form of plug-ins. It is platform independent as it is implemented in Java, and can be downloaded free of charge. The main function of ProM visualizes data and discover a process model. In this BPI 2017 analysis, we used ProM version 6.6 and 5.2.

2.2 Celonis

Celonis also applies process mining techniques to show various digital traces in the process. It shows all process variants, from the most frequent process sequence to the complete visualization of all processes. With the intuitive Process Explorer, as well as many other flexible visualization and filter options, social network analysis.

2.3 Excel

We used Excel (Microsoft Office 2016) to manipulate data. In many cases, we used Excel alongside Dicso, which helped us visualize and refine observations in real time. Excel was especially helpful for performing basic and advanced mathematical functions and data sorting, two capabilities notably absent from the Disco application. The Excel was comfortable because most tools could use the Excel data format. That is, excel allows .csv and .xls file without any transforming.

3 Understanding basic data information

It is important to understand the raw data before we analyze the whole process because it helps to recognize the overall data set and to adopt what kind of analysis. In this part, we briefly search about the raw data before we start the main analysis.

3.1 Raw data

Raw data for this study contains information about the loan process. Activity record started at January 1, 2016 and finished at January 1, 2017. Raw data includes case ID, activity, resource, start/complete timestamp, application type, loan goal, request amount, event ID, Firstwithdrawal amount, monthlycost etc. There are 26 activities done in the process (Fig. 1). Each activity belongs to the activity group. There are 3 groups, A_ group, O_ group and W_ group. Each group represents the state of application, offer and work item. The meaning of each group and activity are summarized in Fig. 1.

Activity	Explanation
A_Group (10)	**A process indicating the status of the application for a bank loan**
A_Create Application	Activity to fill out the application
A_Submitted	Activity to submit the application
A_Concept	Evaluating the application made automatically by the bank
A_Accepted	Step the customer waits for the end of the application
A_Complete	Through an application
A_Validating	Assessing the validity of the application
A_Incomplete	Steps for the unsuccessful application
A_Cancelled	Cancel the application
A_Pending	Wait to receive a loan
A_Denied	Reject the application
O_Group (8)	**A process indicating the offer status**
O_Create Offer	Create offer
O_Created	Completion of creating offer
O_Sent (mail and online)	Send offer to mail and online
O_Sent (online only)	Send offer online
O_Returned	Responses to offer accepted from applicant
O_Accepted	Successfully accepted offer
O_Cancelled	Offer canceled
O_Refused	Offer refused
W_Group (8)	**Process indicating the status of the work item that occurs during the application approval process**
W_Call incomplete files	Collecting applications that did not finish successfully
W_Handle leads	Action on incomplete application
W_Assess potential fraud	Explore whether the application is fraudulent
W_Shortened completion	Decrease application step
W_Personal Loan collection	Additional personal information collection
W_Complete application	Pass the application
W_Validate application	Validity of application
W_Call after offers	Workitem after the offer is sent where the customer is reminded after the offer

Fig. 1. Meaning of the activity group and activities

3.2 Investigation of event logs

The event logs consist of 31,509 cases with 561,617 events and 4,047 variants. The range of activities included in a case are 8 to 61. The average number of activities is about 17.8 and most of cases include 11 to 19 activities. About 72% variants have just one case. Top 1 to 40 variants have 20,430 cases which are 65% compared to the number of whole cases.

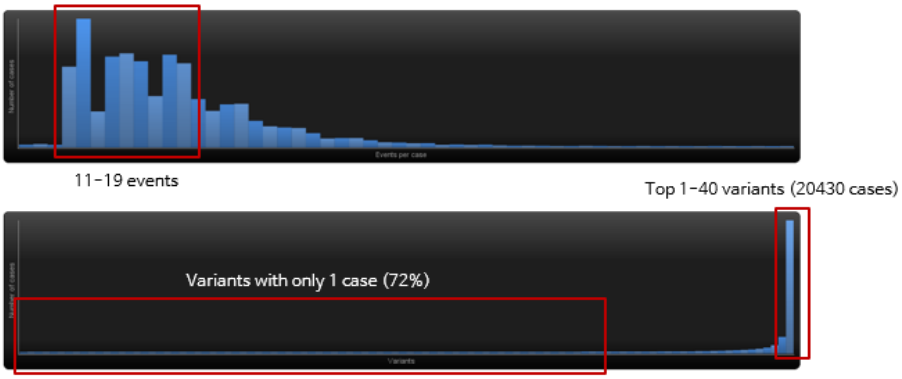

Fig. 2. Events per case (above), case variants (below)

Fig. 2 represent graphs from Disco. The above graph shows events per case and below graph shows case variants. The variant that includes the maximum number of cases has total 3,656 cases and each of case has 12 activities. We observe that the less variant includes cases, the more the number of activities that include a case is increasing. From the result, the basic process has 11 to 19 activities and sometimes a process should have more activities if they are needed.

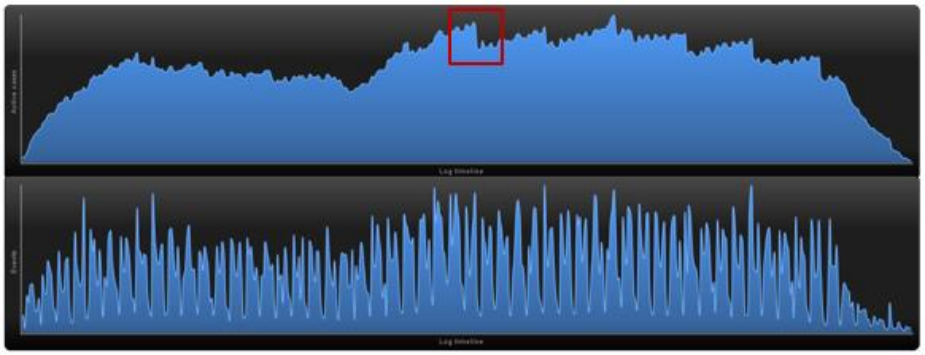

Fig. 3. Active cases over time (above), event over time (below)

We found some characteristics from Fig. 3. In Fig. 3, the above graph represents the active cases over time and the below graph represents activities over time. We

recognized that there is a monthly drastic reduction of active case between day of 22 and 23 in the above graph. Furthermore, there is a reduction of activities per 7 days in the below graph. Matching with the calendar, we found that the reduction point was every Sunday.

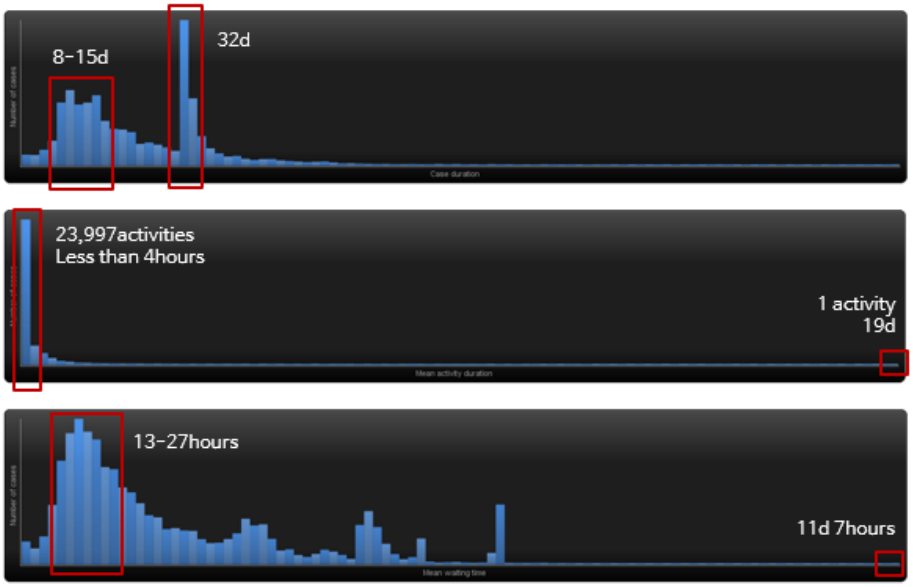

Fig. 4. Case duration (top), Mean activity duration (middle), event per case (bottom)

Next, we research about each of case and time duration (Fig. 4). The top graph represents case duration and middle graph represents mean activity duration, and the last graph shows mean waiting time. Through the top graph, we could find most cases ended up within 35 days and there were cases that ended up within 32

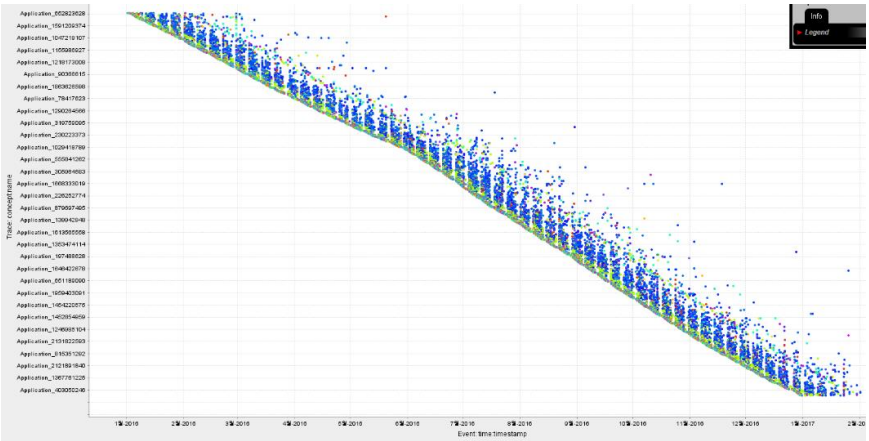

Fig. 5. Dotted chart (actual time-case, color: activity)

days or 33 days. Refer to middle graph and bottom graph, it is known that the case duration was more influenced by waiting time than activity duration.

Fig. 5 is the dotted chart using ProM. In the dotted chart, X-axis represents the actual time of activity and Y-axis represents the case. The color of dot means category of activity. We found that few activities were done in a certain period from the dotted chart. This was Sunday that we already check in previous dotted chart. We also found that the O_ labeled activity was not performed on Sunday. The O_ labeled activity was performed a certain weekday time: that was about 8:00 A.M to 9:00 P.M.

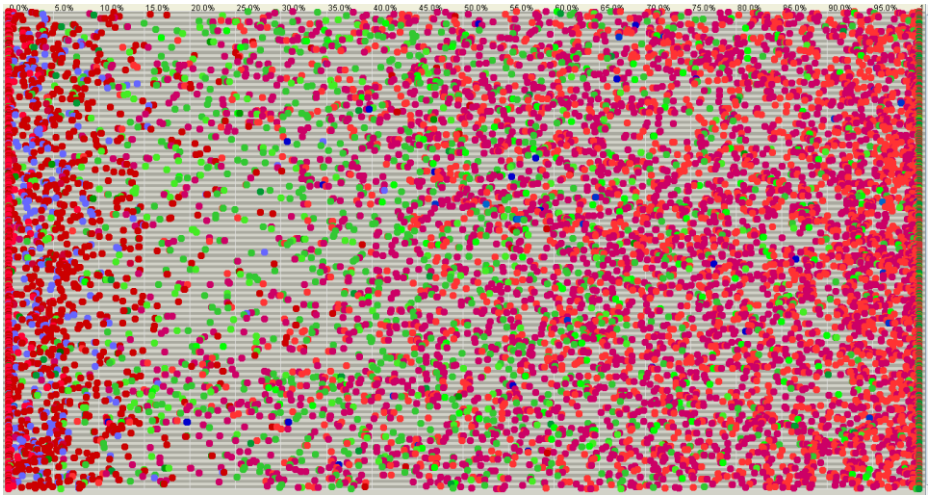

Fig. 6. Dotted chart (relative time-case, color: activity)

From the dotted chart (Fig. 5), we substitute X-axis as relative time and color attribute as activity. The result is shown in Fig. 6. To observe a pattern, we set up A_ labeled activity as red, O_ labeled activity as green and W_ labeled activity as blue each. After applying the setting, we found that when a process start, there were red dots and blue dots follow red dots. There are green dots in the middle of the process. The density of green dots is low which means that the waiting time is longer than other activities.

3.3 Investigation of whole activity

To analysis process, we investigated each of activity. Table 1 shows activity with frequency in descending order. From the Table 1, it is figure out that there are activities with same frequency such as O_Create offer/O_Creat, A_Create Application/A_Concept/ A_Accepted. It means that activities with the same frequency always occur at the same time, or when one activity occurs, another activ-

ity necessarily occurs. In addition, most activities have zero value for median duration and mean duration. In case of activities that non-zero value, there was substantial difference between median duration and mean duration. These activities might have some outliers. In order words, the distribution of duration might not follow the normal distribution. Especially only W_ labeled activities had non-zero median/mean duration value, so A_ labeled and O_ labeled activities were considered as just indication that represented the completion point of the activity. We remarked that the W_Assess potential fraud activity could cause the bottleneck because the median and mean duration were about 15 hours. On the other hand, in case of W_Handle leads, W_Complete application activities might not affect the whole process duration time because their median and mean duration were short.

Table 1. Summary of activity information

Activity	Freq.	Median duration	Mean duration	Start/End
O_Create Offer	42,995	0 millis	0 millis	
O_Created	42,995	0 millis	0 millis	
O_Sent (mail and online)	39,707	0 millis	0 millis	End
W_Validate application	39,444	0 millis	23 hours, 1 min	End
A_Validating	38,816	0 millis	0 millis	End
A_Create Application	31,509	0 millis	0 millis	Start
A_Concept	31,509	0 millis	0 millis	
A_Accepted	31,509	0 millis	0 millis	
W_Call after offers	31,485	0 millis	23 mins, 23 secs	End
A_Complete	31,362	0 millis	0 millis	End
W_Complete application	29,918	7 mins, 23 secs	6 hours, 4 mins	
O_Returned	23,305	0 millis	0 millis	End
W_Call incomplete files	23,218	0 millis	21 hours, 11 mins	End
A_Incomplete	23,055	0 millis	0 millis	End
O_Cancelled	20,898	0 millis	0 millis	End
A_Submitted	20,423	0 millis	0 millis	
O_Accepted	17,228	0 millis	0 millis	
A_Pending	17,228	0 millis	0 millis	End
A_Cancelled	10,431	0 millis	0 millis	End
O_Refused	4,695	0 millis	0 millis	End
A_Denied	3,753	0 millis	0 millis	End
W_Handle leads	3,727	1 min, 24 secs	20 mins, 58 secs	
O_Sent (online only)	2,026	0 millis	0 millis	End
W_Assess potential fraud	355	15 hours, 33 mins	3 days, 1 hour	End
W_Shortened completion	76	0 millis	0 millis	End
W_Personal Loan collection	4	0 millis	0 millis	End

For the next step, we analyzed start and end activities. At first, Disco was used for this analysis. However, it was hard to know the sequence of activities that are

recorded at the same time. For example, in many cases, O_sent, W_call after offers, and A_Complete had the same starting timestamp. Therefore, these activities were considered as end activities only in the ProM.

The whole cases started with A_Create application. In order words, there was only one start activity. When we analyzed the data, 17 activities were candidate of end activity. We assumed that some candidates were not actual end activities in the process for some reasons.

3.4 Investigation of end activity

Clarification of start and end activity of process is important to find incompletely recorded cases. Incompletely recorded cases make noise that may reduce accuracy of analysis. In this study, we assumed that there were problematic data that were not recorded properly or not completed for some reasons. For example, some cases continued after the record finished. Some cases might be missed during the process. We did the analysis to clarify actual end activity. First, we arranged each of case that has different end activity (Table 2).

Table 2. Summary of end activity

End Activity	Case frequency	Cases (%)	Events (%)
A_Cancelled	161	~1	~1
A_Complete	30	~1	~1
A_Denied	34	~1	~1
A_Incomplete	53	~1	~1
A_Pending	12791	40	41
A_Validating	11	~1	~1
O_Cancelled	19712	46	45
O_Refused	4693	11	11
O_Returned	2	~1	~1
O_Sent (mail and online)	46	~1	~1
O_Sent (online only)	13	~1	~1
W_Call after offers	2	~1	~1
W_Personal Loan collection	2	~1	~1
W_Shortened completion	1	~1	~1

From Table 2, all of end activities are included the cases which are less than 1% except A_Pending, O_Cancelled, and O_Refused. Considering the meaning of each activity, total 5 activities: A_Pending, O_Cancelled, O_Refused, A_Cancelled, and A_Denied can be the end activity. Other activities can't be the end activity considering the meaning of each activity therefore these activities are not end activity, but just outliers that have problem on recording or process. To recognize this outliers, each of activity should be included in process (setting as mandatory) to analyze using the DISCO. As a result, we found that A_Cancelled and

A_Denied directly followed O_Cancelled and O_Refused (Fig. 7). This phenomenon is thought to occur because the event log contains both the activities of the applicant side and the offer side. In order words, from applicant view, A_Cancelled and A_Denied activities are the final activities in the process, but from offer view, these activities are not the final activities. So, considering the meaning of A_Cancelled and A_Denied, it looks like end activity, but the corresponded offer side activity should become end activity in the actual process, not these activities. Likewise, when we observed the offer log, the 40% cases ended with O_Accepted. But after O_Accepted was performed, A_Pending activity was performed in the whole event log.

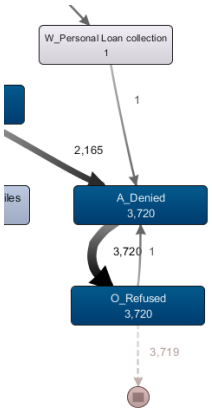

Fig. 7. Cases including A _Denied

In conclusion, we determine the three activities as end activities: A_Pending, O_Cancelled, O_Refused. The A_Pending activity means that loan was completed and the two other activities means that loan application was refused. To increase the accuracy of analysis, the following analysis was performed using filtered data that have only the data with the three activities as end activity. Event logs with three activities as end activity are more than 99% of the whole event log. The 98 cases were excluded. So 31,411 cases, 560,023 events were selected for final analysis data. In case of variants, 50 variants were excluded and leaving 3,997 variants.

4 Understanding main process

The data used in this study were provided by Dutch bank. Therefore, the most of workflow process is related to banks usually do. Especially this process is concerned with the loan validating. The activity in process can be divided into three parts.

First A_ labeled activities mean that the state of loan application. A_ labeled activities consist of decision making for loan validating and making loan application. Next, O_ labeled activities are related to offer to customer from bank. O_

labeled activities include creating offer, sending the offer to the customer, and ultimately receiving the customer's response to decide whether to accept or cancel the offer. Finally, W_ labeled activities are related to additional activities that occurred in the loan approval process. These activities should be divided that performed by applicant or performed by user.

4.1 Investigating through offer log

To investigate the whole event log, we first observed the offer log. By discovering the offer log process, we could understand offer-side process and it would help us understand overall event log. In the offer log, the end activities were O_Cancelled, O_Accepted, O_Refused. Using Disco, we could see the helicopter view for the offer process (Fig. 8). The left figure represent the case frequency of activity and right one shows median duration. There are some characteristics in the offer-side process. First, O_Created should be followed by O_Created offer. This supports the fact that the frequencies of the two activities are the same in the whole event log (Table 1). Second, all cases can be classified into cases that pass O_Returned and those that do not. In addition, O_Accepted can be seen as an activity performed only when O_Returned occurs. O_Returned is the activity corresponding to the response, which means that the application has been accepted on the offer side. Therefore, O_Accepted can only be performed if the application is successfully accepted. In the case of O_Cancelled and O_Refused, the case can be cancelled or rejected regardless of whether or not the application has accepted it on the offer side, so it can occur regardless of whether O_Returned has been performed. Finally, we can see that bottleneck occurs after O_Sent from the right figure. Especially, the duration is longer for the cases with both mail and online than only online. If this O_Sent is randomly allocated, time duration will almost same. Therefore, it can be expected that the activity will only be performed online when the applicant or process meets certain conditions. The time from O_Sent to the next activity was the longest when O_Cancelled followed, and the shortest when O_Returned followed.

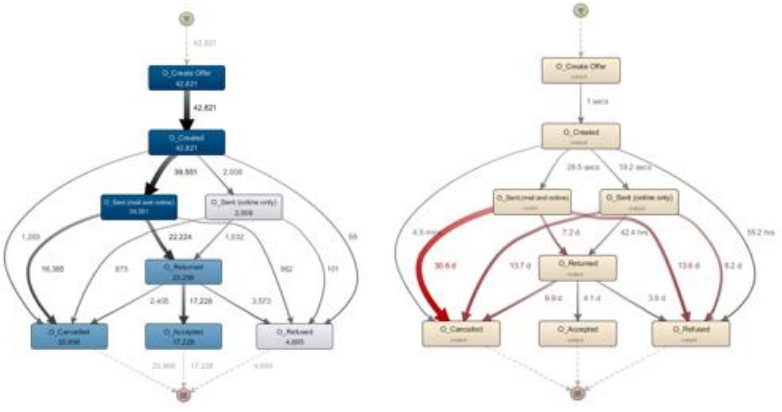

Fig. 8. Case frequency (left), median duration (right)

4.2 Investigating whole event log

To understand the whole process, we discovered the process. First using DISCO, we recognized the overall process. Then we discovered petri net using alpha algorithm provided by ProM.

4.2.1 Using Disco

Since we aimed to understand the approximate flow of the process through Disco, we looked at the process of variants that we think could represent the whole case. At this time, the variant was selected for each end activity, considering that the process would be different depending on the end activity of the process. Selection criteria are as follows. Since the process of the selected variant should be able to represent the entire case, it should include many cases. In addition, variants with a large number of activities were excluded from the selection. If the number of activities is more than 18, it is considered that the main process will not be represented properly because there will be loop or there is a possibility that the activity has gone through additional activity. As a result, the flow represented by the three selected variants was observed and then integrated.

The variant that has O_Cancelled as end activity includes total 3,655 cases with 12 activities. The selected variant took up 25% that has O_Cancelled as end activity and have the most number of cases. The variant that has A_Pending as end activity includes total 1,454 cases but it has 19 activities. The variant has 1,367 cases which are the second most number of cases and has 15 activities. There is no big difference between the two variant, we selected variant which had less activities. The variant took up 11% that have A_Pending as end activity. In the last part of this analysis, the variant that has O_Refused as end activity includes 719 cases and consists of 15 activities. The variant that has O_Refused as end activity takes up 19% that has O_Refused as end activity. Seeing as helicopter view each of variant, there is a common process except 5 to 6 activities. The result is shown in Fig. 9.

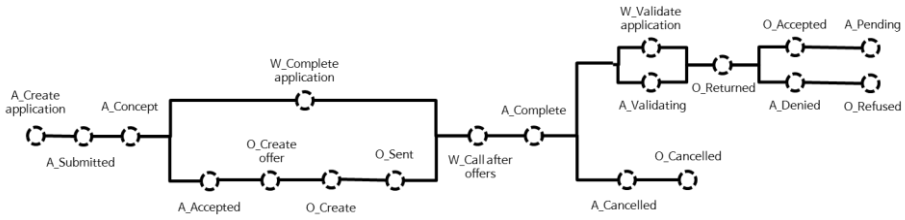

Fig. 9. Main process of event log

This shows that most cases go through a similar main process before A_Complete. After the A_Complete is done, the process goes through and ends with different activities. Since the result is an approximate process through the selection of some data and the helicopter view, it is difficult to say that it reflects the whole

event log properly. Therefore, in this study, we examined whether the process reflects actual event log through ProM.

4.2.2 Using ProM

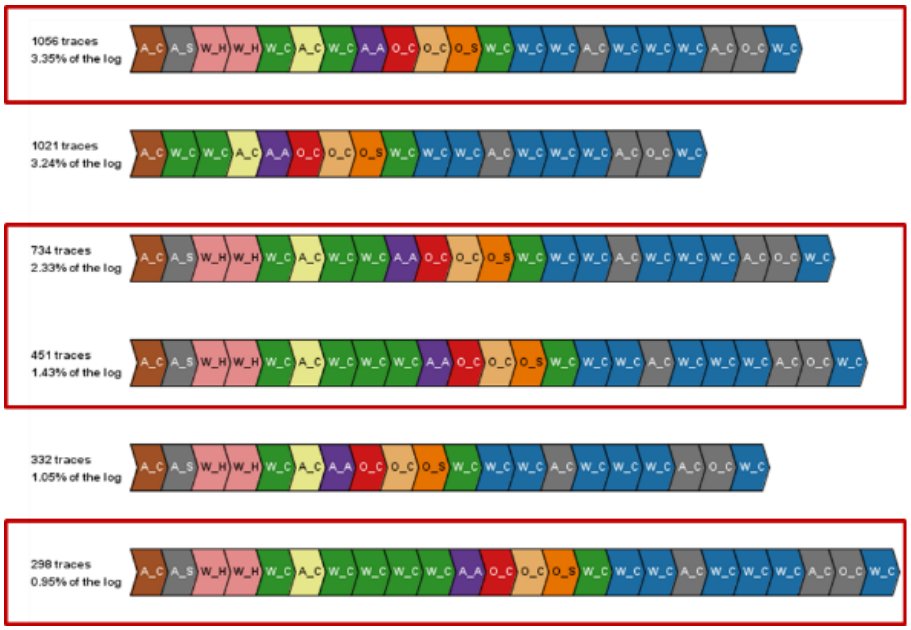

Fig. 10. Log trace (ProM 6.6)

ProM 6.6 provides function to view event log flow by variant at a glance. First, we looked at the trace to discover which process is generally happening. At this time, the event log was also classified according to the end activity. As a result, we found that there are many self-loops in the process. Many cases that are classified as other variants even though they actually go through the same process (Fig. 10). Although they go through same activity order, variants 1, 3, 4, and 6 in the Fig. 10 are considered as different process because of self-loop. Therefore, it is thought that it would be easier to discover the main process by removing the self-loop. Moreover, event logs containing self-loops can not be detected on petri net, making it difficult to do accurate analysis. Therefore, we removed the self-loop from all event logs. At this time, the time stamp of the activity to be integrated is set to the start time and the end time of the self-loop. Then, variants containing more than 1% (310) of the total cases were selected for analysis to prevent generation of complicated petri net. The filtered data shows a much clear pattern than before. As a result of extracting petri net of filtered data, there was an optional process depending on end activity. However, the entire process was similar to the process derived from Disco analysis. Fig. 11 is petri net with O_Refused as end activity with fitness 0.86. The petri net is very similar to the previous process (Fig.

9), so it can be said that the process suggested in Fig.9 represents the main process of the entire event log.

Fig. 11. Petri net with O_Refused

4.3 Interpretation of derived process

The derived main process can be interpreted as follows. First, when the applicant creates an application (A_Create application) and submits (A_Submitted), an activity (A_Concept) is automatically performed to confirm the application. Thereafter, the customer waits (A_Accepted) while passing through the W_Complete application, and the bank generates an offer (O_Create offer, O_Create) and sends it to the customer (O_Sent). If both the bank and the applicant confirm the progress (W_Call after offers, A_Complete), they will perform different end activities according to the applicant's next activity. If the applicant cancels the application at this stage (A_Cancelled), O_Cancelled is performed as an end activity. If the customer does not cancel the application, the bank will evaluate the application (W_Validate application) while the applicant will review the offer (A_Validating). When this process is completed (O_Returned), the applicant rejects the offer (A_Denied → O_Refused) or accepts the offer and waits for the loan (O_Accepted → A_Pending).

5 Understanding user

There were 145 users in the raw data. It was important to know each user has any rules and has any team structure and the difference for the performance time. It might help to improve the process improvement.

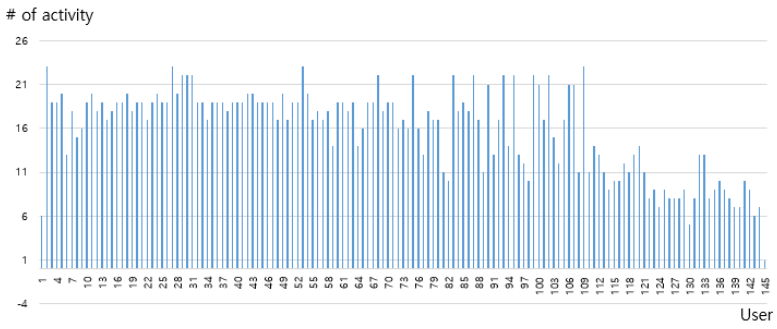

Fig. 12. The number of activity per user

5.1 Basic information

Fig. 12 represents each user performed the number of activities. Fig. 13 shows the number of users according to the number of activities performed. There are a total of 26 activities. One user performs an average of 16 activities. The user who performs 19 activities has the most number (33 users). There were no users performing more than 24 activities. Among users with user numbers greater than 120, the percentage of users performing less than 10 activities was high. In particular, user 145 was only performing one activity, and the activity is a W_Personal Loan collection.

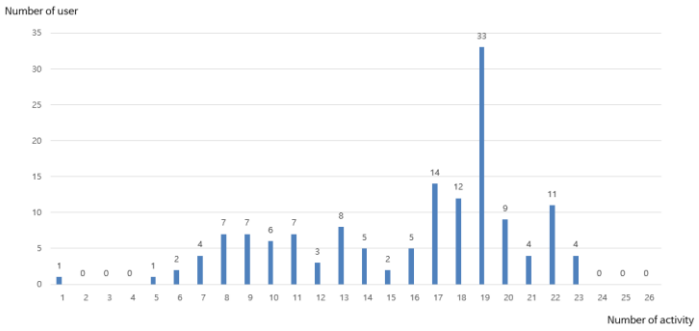

Fig. 13. Number of user depending on the number of activities to perform

We divide the activity into activities starting with A_, O_, and W_, and shows the ratio of each activity group to the total activity performed by each user (Fig. 14). It can be seen that for most of user, the ratio of A_ group is 40% and O_, W_ group is 30%. However, there were 8 users who could be seen as outlier (Table 3).

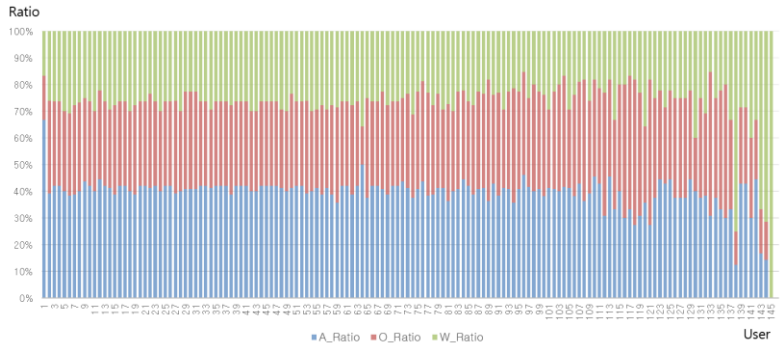

Fig. 14. Ratio of activity group for each user

User 1 occupied 67% of the activity of A_ group. User 118, 121, and 133 were performing more than 50% of O_ group activities. User 138, 143, 144, and 145 had a high rate of activity beginning with W_ group.

Table 3. Outlier for ratio of activity group

User	A_ratio (%)	O_ratio (%)	W_ratio (%)	Number of activity
1	67%	17%	17%	6
118	27%	55%	18%	11
121	27%	55%	18%	11
133	31%	54%	15%	13
138	13%	13%	75%	8
143	17%	17%	67%	6
144	14%	14%	71%	7
145	0%	0%	100%	1

Fig. 15 shows the number of users performing each activity. Most activities are performed by more than 100 users. However, some activities such as A_Pending, O_Retruned, and O_Accepted are performed by less than 50 users. In particular, A_Submitted and W_Personal Loan collection are performed by one user respectively. It is thought that there is likely to be bottleneck for activity with a small number of users. However, there was no significant relationship between the number of users and the execution time of the activity.

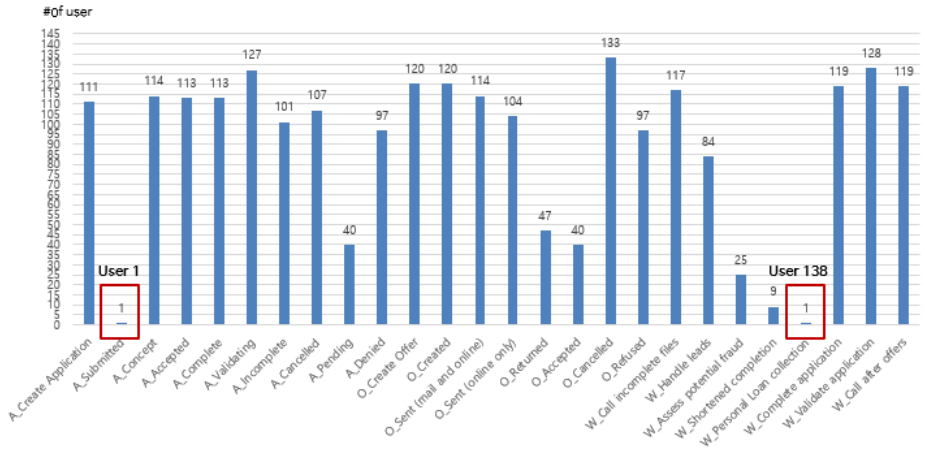

Fig. 15. The number of user per activity

5.2 Find team/functional structure

In this study, for a broad understanding of the whole loan processes, we tried to find the users' team structure and function structure. To do this, we applied various functions provided by ProM, but could not get meaningful result. The following analysis was conducted by judging that it would be helpful to construct the

structure by directly investigating the attributes of the event log. First of all, the user distance according to the activity was calculated to classify the users who have similar activities. Based on user distances, we tried to find handover rules. Next, we tried to group users by considering the activity execution time. Finally, except activity and time, we tried to classify the users through the characteristics given in the event log (loan goal, request amount, etc.).

5.2.1 Analysis of the users who take charge of similar activities

First, the Euclidean distance was calculated after displaying the activities performed by each user, and hierarchical clustering was drawn using R program (Fig. 16).

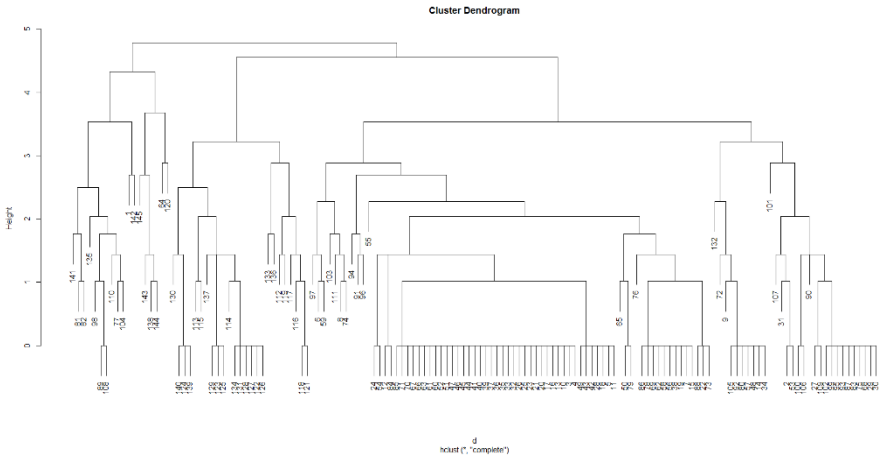

Fig. 3. Hierarchical clustering based on Euclidean distance

Users with distance 0 means users with the same activities. For example, user 89 and user 108 have the same activity set. In this study, dotted chart provided by ProM were used to find out whether users who perform the same activity actually have a similar work pattern. To detect only target users, all users are set to blue, and the color of the target user is set to a different color. As a result, the following facts were found.

First, when set logical relative as the x-axis, it figures out that the activity is performed in a similar order because a user group with a distance of 0 performs the same activity (Fig. 17). For example, user 22, 73 and 88 perform activities corresponding to the front of the entire process. In the case of a group performing an activity located in the front, the activity is performed at a similar time in the most processes. However, in the case of a group performing activity at the back of the process, such as user 124, 139, and 140, depending on the number of activities that process goes through, there is a slight difference in the timing of the activity.

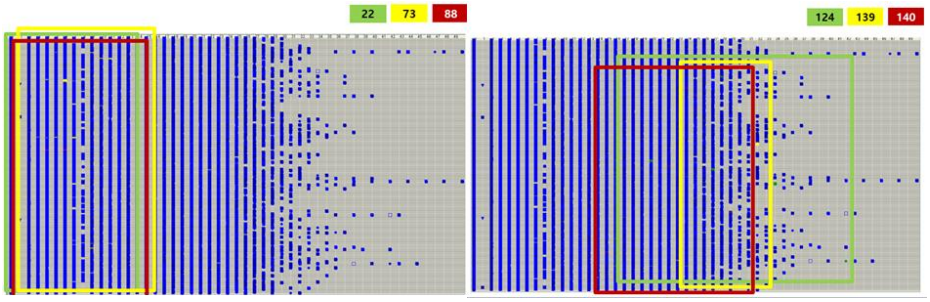

Fig. 17. Dotted chart (relative time-case, color: user)

If the x-axis of the dotted chart is set to actual time, there is a difference in the work period between users who perform the same activity (Fig. 18). In the left dotted chart, although three users perform the same activity, they don't actually work at the same time. As a result of analyzing the work time of each user in the event log, user 124 performed activities from May 3, 2016 to July 28, user 139 performed activities from August 15, 2016 to January 26, 2017 and user 140 performed activities from November 11, 2016 to January 10, 2017. This result is slightly different for the result on the dotted chart. Because the dotted chart shows a lot of activity, some of them are hard to find. Unlike user 124, 139, and 140, who work only for a limited period of time, there are also users who perform activities in all periods during data collection. Typically, user 100 (yellow dot) in the right dotted chart is included in almost all periods in the dotted chart. User 100 performed the activity from January 8, 2016 to January 27, 2017. On the other hand, user 106 started to perform activity on September 29, 2016, despite performing the same activity as use 100.

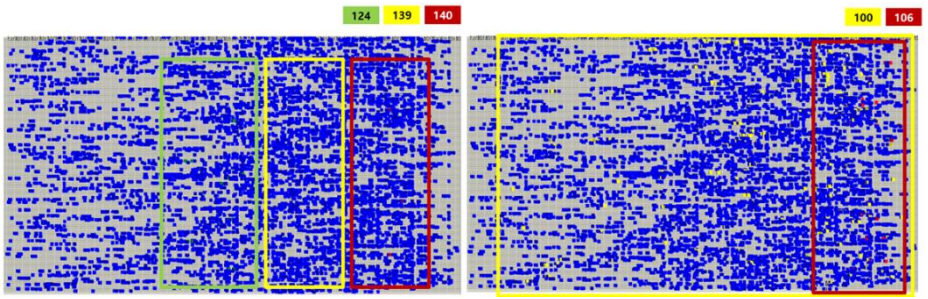

Fig. 18. Dotted chart (actual time-case, color: user)

5.2.2 Find Rules of Handover

We tried to find the rule of handover after finding a group of users who performed the same activity, there were no significant result using ProM. Therefore, in this study, we tried to get some patterns by selecting some event logs. First, the cases that go through the same step were analyzed to find the rules for users who work together when going through the same step. To do this, variants containing the largest number of cases were selected. As a result, 3,656 cases with 12 activities were selected. As a result of analyzing the user who performed the selected process, three patterns were found (Fig. 19). Fig. 19 shows the activity performed sequentially from the top, and the cell color indicates the user. There were certain rules for activities that were performed by different users in one case, but we could not find a pattern such that a particular user worked with another user. Therefore, we conclude that handover rules between users are hard to find. Instead, the analysis showed that the time duration of the activity is affected by the type of activity, not by the responsible user. So, the time required to perform a specific activity is not affected by which user performs.

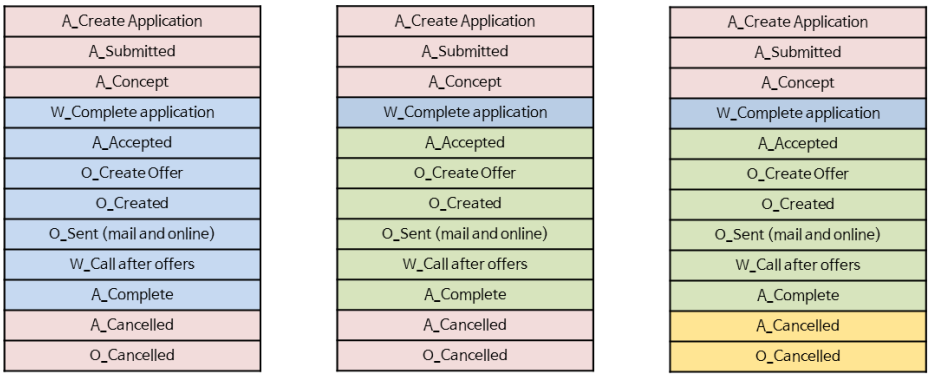

Fig. 19. User pattern of variant 1

5.2.3 Analysis of activity execution time

In this analysis, it was concluded that the previous conclusion that the time taken for the activity is not dependent on the user was likely to be limited to only one variant. In addition, since most of the time durations of activities are close to zero, we think that we need a deep analysis of activities which time duration is not zero. Therefore, the activity execution time according to the user for the activity with the long time duration were analyzed. We analyzed the W_Assess potential fraud and W_Complete application which has the longest median duration among 26 activities. For the analysis, only the Case ID, Resource and Timestamp of those activities were selected from the whole event log. The activity execution time of each newly created activity set by user though DISCO was analyzed. In this case, if the activity was performed several times in one case, we examined whether the time duration differs according to the order of execution.

The results of analysis of W_Assess potential fraud are as follows. The W_Assess potential fraud is performed 354 times in total 300 cases, and the duration of activity is the longest with three-day duration. The maximum number of W_Assess potential frauds performed in one case was 3. In this case, the total activity was performed by 19 users. Several patterns were found through analysis. First, if the cases that the activity is not repeated (activity frequency = 1), the median duration of the remaining 16 users except the user 138, 143, and 144 is zero. Those three users whose time duration is not zero have a significant difference in median duration which is more than one day. Second, in single case where activity is performed twice, the time duration for the first execution is long and the time for the second execution is zero. It is expected that if the activity is performed more than twice, it would utilize the previous data. However, most of the first activity in this case was performed by user 138, 143 and 144. Therefore, as in the case of activity frequency one, only the time durations of the corresponding users were significantly longer for those users. (Fig. 20). Finally, even if the activity frequency is three, the time durations of users 138, 143 and 144 are relatively longer than those of other users, but the number of cases is only six. Therefore, it is difficult to derive a meaningful result.

Resource	▲ Frequency	Relative frequency	Median duration	Mean duration	Duration range
User_144	36	42.86 %	1 hour, 39 mins	1 day, 8 hours	13 days, 1 hour
User_138	21	25 %	18 hours, 12 mins	1 day, 18 hours	11 days, 4 hours
User_143	12	14.29 %	3 hours, 28 mins	2 days, 13 hours	12 days, 33 mins
User_29	3	3.57 %	0 millis	20 secs	1 min
User_68	2	2.38 %	30 secs	30 secs	1 min
User_75	2	2.38 %	0 millis	0 millis	0 millis
User_102	1	1.19 %	0 millis	0 millis	0 millis
User_99	1	1.19 %	1 min	1 min	0 millis
User_93	1	1.19 %	2 mins	2 mins	0 millis
User_54	1	1.19 %	0 millis	0 millis	0 millis
User_106	1	1.19 %	0 millis	0 millis	0 millis
User_83	1	1.19 %	2 mins	2 mins	0 millis
User_30	1	1.19 %	0 millis	0 millis	0 millis
User_137	1	1.19 %	0 millis	0 millis	0 millis

Fig. 20. Time duration of each user for W_Assess potential fraud

Next, the W_Complete application is executed 29,368 times in a total of 29,638 cases. Unlike W_Assess potential fraud, W_Complete application is an activity that goes through most cases. This activity is performed by a total of 119 users. The mean duration is about 6 hours, which is the second longest in the activity. The maximum number of times a W_Complete application is executed in single cases is 12. In the case where this activity is performed several times in single case, the time duration of all the activities except the first activity is zero. But no order or rule could be found among the users who performing this activity. In the case where this activity is performed only once, there are 3 users (user 112, 120 and 145) with time duration as zero. But, the user with the longest median time takes 19 minutes of execution time unlike the W_Assess potential fraud, so that the deviation between users was not large. However, if the activity is performed several times in the case, there are users with remarkably long time durations such as user 23, 26, 51, 79, 92, 105 and 132. Most of them were time duration as zero.

From the above results, users with significantly longer time durations than other users could be derived (User 23, 26, 51, 79, 92, 105, 132, 138, 143, 144). Also, there are many users whose activity time is zero. Therefore, in this study, if there is a user with all activity time as zero, it means this user would be system rather than person. The reason for this is that if there is a process carried out by the system during the loan, it is determined that there will be no work to be done through calculation that takes a very long time. So, we wanted to find a user whose activity execution time is close to zero. However, we could not find a user that satisfies these conditions. On the other hand, there was an activity with zero execution time for all users. Therefore, the users in the event log determined that they are all people, not system. In the case of an activity whose execution time is zero, it can be expected that it is only a step to check that the activity has passed.

Even if user perform the same role, there is a difference in time duration for each user. Therefore, we conclude that it is difficult to find the user's team or functional structure through time duration.

5.2.4 Classification of the user's role based on characteristics except activity and time.

In this study, we analyze event log by assuming that the users to be allocated would be different according to the various characteristics except activity and time, which are known from the event log. The cases were divided according to each condition to find out whether there are any differences in the users who perform activities according to the loan goal, request amount and so on. For example, we divided the cases into groups of cases which follow the same steps but, with a request amount as 5,000, and 10,000 to analyze which users perform each case groups. As a result, it was not possible to derive meaningful results from all characteristics. We decided that this result did not take into account the order in which the activity was performed. We divided the cases according to each condition for the variants with the largest number of cases selected. However, we could not find any significant user structure. Therefore, in this study, we conclude that all users will be assigned randomly regardless of cases attributes. In conclusion, we could not find a meaningful user structure other than the group of users performing similar activity through the provided event log.

6 Answers for each question

This section presents the analysis and answers for the three questions from the BPI Challenge 2017.

6.1 Question 1

Question 1 is as follows.

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.

The question 1 is to identify whether the throughput time of the process is caused by the user or by the applicant. In general, throughput time means the time interval between input time and output time. In this study, throughput time is analyzed in terms of process and activity. First, from a process perspective, the throughput time is the time taken from an activity to the next activity, that is, the time spent on an arc in the petri net. The throughput time in terms of activity is the difference between the starting time and ending time of the activity.

In this study, the throughput time that occurs when a user goes through a general process were analyzed. First, the median, mean, and max time for each activity and arc were recorded. As a result, the time spent in the arc was relatively long compared to the time spent in the activity. Therefore, we determined that the throughput time of the whole process will be determined by the time spent in the arc. We recorded the time required for all the arcs in two activity combinations and selected the arc whose mean time was more than 1 day. Based on the meaning of each activity and process flow, we determine whether the selected arc is dependent on the user or the applicant. As a result, we analyze the percentage of the throughput of the process by the user or the applicant by summing the time of the arc classified as user and applicant.

First, 12 arcs with significantly long mean time were selected. The results are summarized in the order of total arc time, total, median, mean, max time, arc absolute and case frequency (Fig. 21). From Fig. 21, some information was derived. First, it is expected that the distribution of time does not follow the normal distribution because the difference between median time and mean time is 5, 6, 8, 9, 10 arc. In addition, the max time was longer than the mean time by 3 times, except for 1, 11, and 12. In particular, 2, 3, 4, and 7 arc can be said to have outliers in a small number of specific cases because the distribution of time is expected to follow normal distribution. Therefore, time problems arising in such an arc can be solved by analyzing the cause of outliers.

On the other hand, arc 1, 4, and 11 have a median time of more than 30 days and a mean time of more than 27 days. In most cases involving these arcs, a bottleneck would occur when process go through these arcs. In particular, these arcs represent the time generated by the applicant and are the arc ending with A_Cancelled. This shows that when the applicant cancels the application, the waiting time in the process is long. Finally, the 1, 2, 3, and 6 arcs are the arc that contains at least one quarter of the whole case. Therefore, bottlenecks would occur when many cases pass through these arcs. These arcs should analyze the work of both applicant and offer side when go through it. Time problems can be solved in such

a way as to grasp the overall time-consuming causes such as inefficient work progress and communication difficulties.

	From	To	Part	Total	Median	Mean	Max	Absolute	Case
1	A_Complete	A_Cancelled	Applicant	599.8yrs	30.7d	27.4d	32.6d	8004	8004
2	A_Complete	A_Validating	Applicant	216.2yrs	7.1d	8.7d	30.3d	9073	9073
3	A_Incomplete	W_Validate application	Applicant	96.3yrs	23.2hrs	59.5hrs	13.2wks	14169	10410
4	O_Sent (mail and online)	A_Cancelled	Applicant	78.6yrs	30.7d	27.6d	17wks	1038	1038
5	A_Incomplete	O_Accepted	Applicant	76.8yrs	3.7d	5.9d	51.8d	4763	4763
6	O_Returned	W_Call incomplete files	Offer	74.6yrs	25.7hrs	47.6hrs	12.2d	13744	13465
7	O_Sent (mail and online)	W_Validate application	Offer	67yrs	6.7d	7.6d	41.1d	3220	3147
8	O_Sent (mail and online)	O_Create Offer	Offer	12yrs	3.1d	6.7d	32.6d	654	555
9	O_Sent (online only)	W_Validate application	Offer	109.5mths	45.8hrs	4d	77.7d	843	821
10	O_Sent (online only)	A_Validating	Applicant	54.5mths	49.3hrs	4.4d	27.8d	380	374
11	W_Shortened completion	A_Cancelled	Applicant	44.3wks	30.8d	28.2d	30.9d	11	11
12	W_Shortened completion	W_Validate application	Offer	22wks	6.1d	8.5d	18.9d	18	18

Fig. 21. Problematic arcs and related information

In~~In~~ this study, each arc was classified as either applicant or offer dependent by interpreting the meaning of arc's starting activity and ending activity in the whole process. Based on this, the ratio of the time duration of all the selected arcs generated by whom is shown (Fig. 22). ~~The contribution of~~ T~~t~~he applicant was contributed much more than the offer to the bottleneck occurring in the process ~~is large~~. However, in the case of max time, contribution of offer is relatively big compared to mean, median, and total time. Max time is most likely an exceptional situation in the process. Therefore, this bottleneck can be solved by exploring the cause of the exceptional case occurring in the offer side.

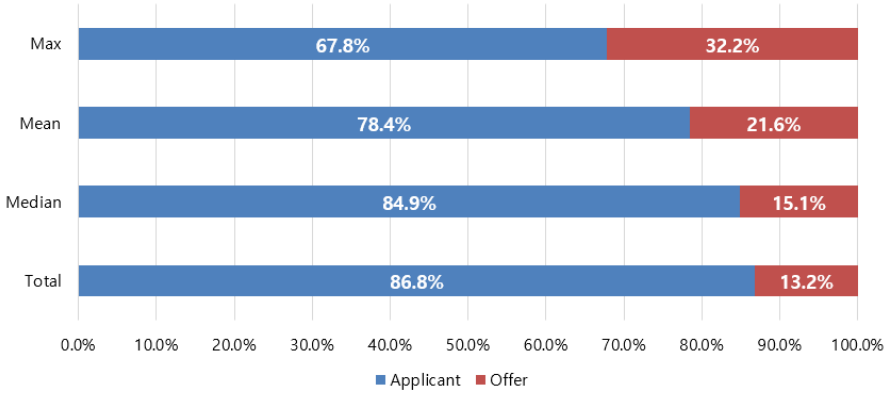

Fig. 22. Throughput time ratio of applicant and offer

In this study, we also investigated the time about the loan goal, request amount, etc., but could not derive a meaningful result related with throughput time.

6.2 Question 2

Question 2 is as follows.

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.

In the question 2, the term 'applicants are confronted with more requests for completion' was used. Therefore, in this study, we assumed that incompleteness is not the incompleteness of the case, but A_incompleteness, which is the activity that occurs when the applicant improperly creates the application. In previous analysis we found that W_Call after offers directly occurse when A_incomplete occurs in the main process which supports the hypothesis of this study. In the question 2, the final outcome can be interpreted in two ways. One is final outcome as the end event that occurred in the process. The other is from the perspective of the applicant, what the state of application ended up with. In this study, we investigated whether the number of A_incompleteness in one case affects the end activity based on two interpretations. For this, the frequency of A_incomplete in each case was counted. After classifying the data with the frequency of A_incomplete, we analyzed the end activity. The results are shown in Fig. 23.

Frequency of A_Incomplete	0	1	2	3	4	5	6	7
Number of case	16,314	9,260	3,941	1,225	350	100	20	7
Number of events	230,252	180,167	94,790	35,066	11,901	3,805	862	346
Number of variants	875	1,143	1,015	980	340	99	20	7
Avg. number of events per case	14.11	19.46	24.05	28.63	34.00	38.05	43.10	49.43
Avg. number of case per variants	18.64	8.10	3.88	1.25	1.03	1.01	1.00	1.00
O_Cancelled = End Activity	10055	2432	1437	526	185	56	11	5
A_Pending = End Activity	3883	5838	2251	628	141	40	9	1
O_Refused = End Activity	2376	990	253	71	24	4	0	1
A_Pending = Forbidden O_Cancelled = End Activity	9357	604	218	66	19	7	0	0
A_Pending = Mandatory	4581	7666	3470	1088	307	89	20	6

Fig. 23. Summary of cases according to the frequency of A_Incomplete

The number of A_incomplete in single case has a range from 0 to 7. The number of cases decreases as the frequency of A_incomplete included in single case increases. In contrary, since the loop containing A_incomplete is repeated several times (Fig. 24), the number of activities included in single case increases.

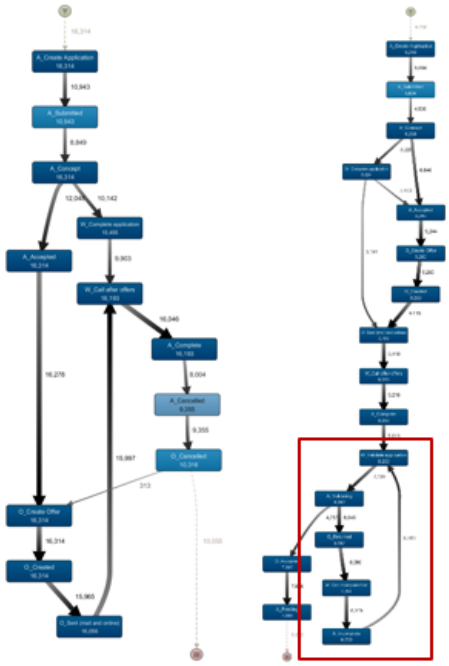

Fig. 24. Helicopter view for A_Incomplete = 0 (left), A_Incomplete > 0 (right)

End activity according to frequency of A_incomplete was analyzed using DISCO. First, we assume that the outcome is the end activity of the process, and filter the end activity of each of the three end activities, respectively, and count the corresponding cases. Next, using this case counting, we interpreted this from the perspective of the applicant. If an offer occurs multiple times in one case, the offer is automatically cancelled when the offer is closed. That is, if A_Pending occurs for one offer, O_Cancelled is canceled immediately after the other offer is canceled (Fig. 25). In this case, O_Cancelled is the end activity in the process but A_Pending is the end activity in the applicant. 4,436 cases have undergone this process, which supports the fact that the time interval between A_Pending and O_Cancelled is very short. In order to reflect this situation, it is necessary to distinguish the case where O_Cancelled is first performed in offer side and the case where O_Cancelled is performed after A_Pending occurs. For the former case, filtering was performed by designating A_Pedning as forbidden and O_Cancelled

as end activity. In the latter case, A_Pending was specified as mandatory and filtering was performed.

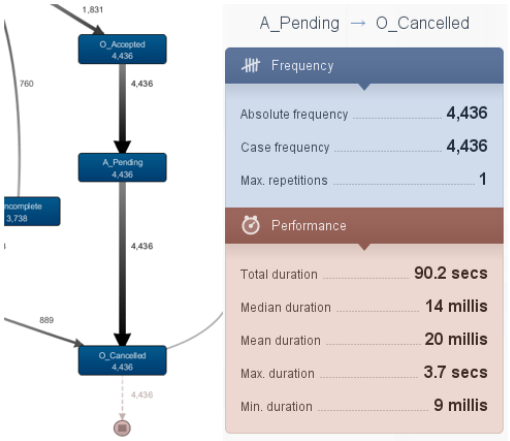

Fig. 25. Time duration of arc A_Pending → O_Cancelled

In order to compare the outcome according to the frequency of A_Incomplete, the frequency of each end activity is converted into the ratio for the whole case, and the result is shown as Fig. 26. First, the end activity in the process side (Ratio of O_Cancelled/A_Pending/O_Refused as End activity) meets the hypothesis of the question. That is, except for the case where there is no A_Incomplete, the rate of ending with A_Pending decreases and the rate of ending with O_Cancelled increases as the frequency of A_Incomplete increases (as requests for completion become more).

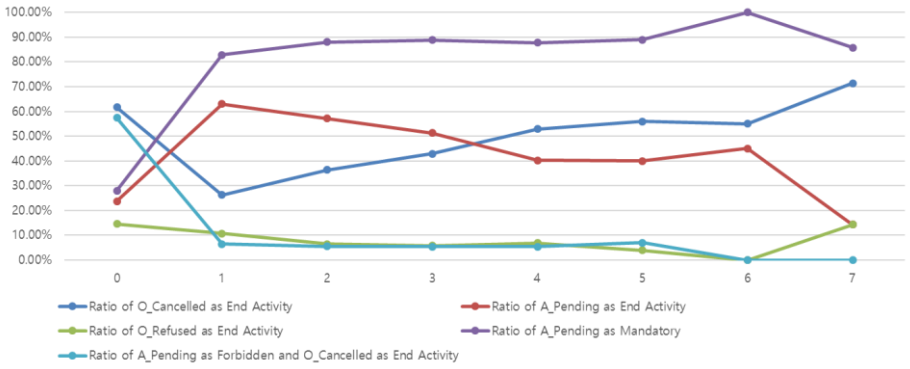

Fig. 26. Ratio of end activity according to the frequency of A_Incomplete

When analyzed in terms of applicants, there are slightly different result existed (Ratio of O_Refused as end activity, Ratio of A_Pending as mandatory, Ratio of

A_Pending as forbidden and O_Cancelled as end activity). If A_incomplete is existed, the probability of pending and the probability of canceling on the offer side hardly change. A case without A_Incomplete is less likely to be pending than a case with A_Incomplete and is more likely to cancle on the offer side. Therefore, when we look at the end activity on the applicant side, the hypothesis is not established.

In this study, the change of the loan goal according to the frequency was also examined. The number of cases where A_incomplete occurred more than 6 times was excluded from the analysis. Except unknown, not specified, and other reasons, the analysis of the top four goals with the most frequent clear loan goals is as follows (Fig. 27).

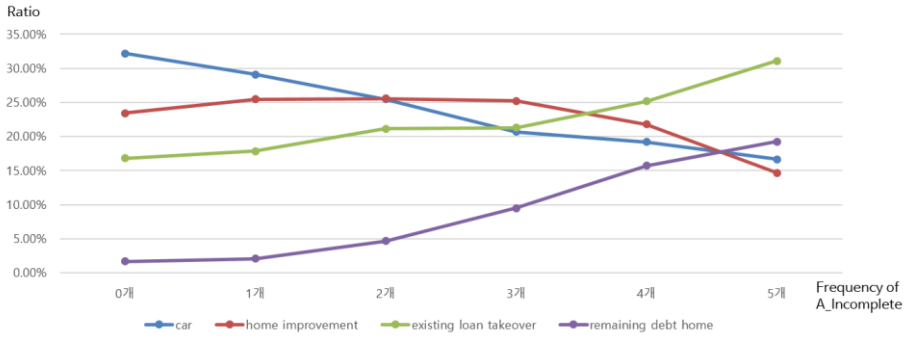

Fig. 27. Ratio of loan goal according to the frequency of A_Incomplete

The frequency of A_incomplete is relatively higher when the loan goal is existing loan takeover and remaining debt home. In contrast, in the case that the loan goal is car and home improvement, A_incomplete takes a relatively small amount and the process ends. The reason is that applicant need to write the application in more detail according to the loan goal, or that the applicant with the specific loan goal has bad credit and needs additional information.

6.3 Question 3

Question 3 is as follows.

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?

The offer refereed in this question is the offer given to the applicant, ie the O_Create offer activity. Conversion is the case where the application is successful, which ended as A_pending. Therefore, the target of question 3 is to find out the

number of O_Create offer in each case, and thus to determine the success and characteristics of the application.

To solve question 3, cases were classified by frequency of O_Create offer in one case. Based on the categorized data, we investigated how many conversations occurred in each case using DISCO. In addition, we analyzed the end activities in a similar way to the question 2. The results are summarized in Fig. 28.

Frequency of O_Create offer	Single Offer	Multiple Offers							
	1	2	3	4	5	6	7	8	9
Number of case	22,750	6,518	1,331	434	124	29	16	12	3
Number of events	363,030	138,445	35,741	13,043	4,376	1,138	708	540	168
Number of variants	878	1,619	893	348	120	29	16	11	3
Avg. number of events per case	15.96	21.24	26.85	30.05	35.29	39.24	44.25	45.00	56.00
Avg. number of case per variants	25.91	4.03	1.49	1.25	1.03	1.00	1.00	1.09	1.00
O_Cancelled = End Activity	7749	5297	1115	380	113	27	13	11	2
A_Pending = End Activity	12177	514	83	14	3	0	0	0	0
O_Refused = End Activity	2824	707	133	40	8	2	3	1	1
A_Pending = Forbidden O_Cancelled = End Activity	7748	2036	314	130	33	27	13	11	2
A_Pending = Mandatory	12178	3775	884	264	83	0	0	0	0

Fig. 28. Summary of cases according to the frequency of O_Create offer

The O_Create offer in each case has a range from 1 to 9. As the frequency of O_Create offer increases, the number of cases decreases and the number of average activities included in the case increased because the loop containing the O_Create offer repeats (Fig. 29).

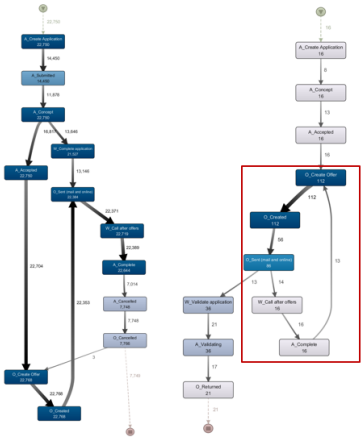

Fig. 4. Helicopter view for O_Create offer = 0 (left), O_Create offer > 0 (right)

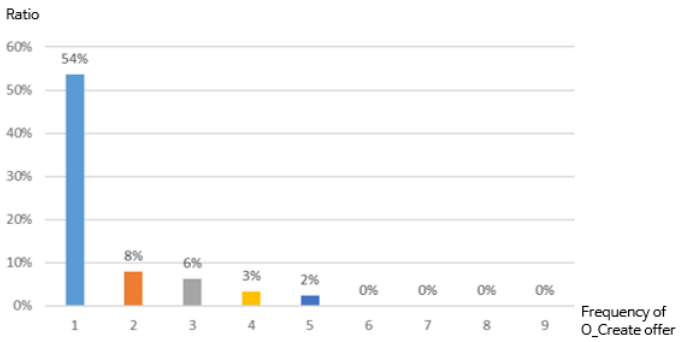

Fig. 30. Ratio of pending case according to the frequency of O_Create offer

Multiple offers occurred in 8,464 cases, which is about 27% of the total cases. The percentage of A_Pending in each cases is shown in Fig. 30. The higher the frequency of O_Create offers, the smaller the pending rate. In addition, we tried to find differences of the frequency of O_Create offer based on loan goal, requested amount, and application type, but there was no significant difference.

7 Conclusion

In this report, we present our findings of the analysis of event logs containing data related with loan application process of Dutch bank, as part of the BPI Challenge 2017. We were also provided three questions related to this process. Before answering the questions, the pre-analysis was conducted which helpful for us to understand the loan application process and answer the provided questions. Firstly, the event logs were inspected by getting to know the basic data information and preprocessing the event logs for analysis. In preprocessing, the cases which did not end up with A_Pending, O_Cancelled, and O_refused were dropped. Secondly, the whole event log and offer event log were investigated to understand the whole process. Thirdly, the analysis about user led to the conclusion that there were no significant team or functional structure. Finally, we have got answers to the three questions as bellows.

7.1 Throughput times

We defined the meaning of throughput time and analyzed it. A total of 12 arcs with significantly long mean time were selected and analyzed. Each arc was classified as either applicant or offer dependent by interpreting the meaning of arc's starting activity and ending activity in the whole process. We also investigated the time about the loan goal, request amount, etc., but could not derive a meaningful result related with throughput time.

7.2 The influence on the frequency of incompleteness

We assumed that incompleteness is not the incompleteness of the case, but A_incompleteness, which is the activity that occurs when the applicant improperly creates the application. In order to compare the final outcome according to the frequency of A_Incomplete, the frequency of each end activity is converted into the ratio for the whole case and analzyed. In addition, the change of the loan goal according to the frequency was also examined.

7.3 The conversion comparison based on number of offers

The target of question 3 is to find out the number of O_Create offer in each case, and thus to determine the success and characteristics of the application. Cases were classified by frequency of O_Create offer in one case. Based on the categorized data, we investigated how many conversations occurred in each case.

This report shows the basic information and process of loan application process and answers three questions provided by BPI Challenge 2017. As we did, the dotted chart analysis and process mining could potentially highlight inefficiencies or bottleneck of user or activity. This kinds of approach could give insights to analysis expert. But when applying our analysis results to real-life process, there could be limitation existed. Because we analyzed the event log without any feedback from the user. Event log has a lot of useful data, but does not contain all of them. So there will be more actual and useful insight with a combination of our analysis results and the person who has domain knowledge.