Root Cause Analysis with Machine Learning in Business Process Management

Samik Real[1] and Jessica Sutanto[2]

[1] Hasso Plattner Institute
Samik.Real@student.hpi.de
[2] University of Potsdam
sutanto@uni-potsdam.de

Abstract. Root Cause Analysis (RCA) focuses on finding underlying problems in processes and eradicating them instead of treating only the symptoms. By using historical data in the form of event logs and combining it with machine learning, it is possible to digest large quantities of data and analyse it through RCA techniques. A structured methodology for applying machine learning for RCA is proposed in this paper. This methodology facilitates event log insight extraction and RCA application. Validation of this methodology is done on a real world data set from a dutch financial institution (4) where problems such as process delays are diagnosed and possible solutions are proposed. Guidelines for feature engineering for RCA is also presented in this paper.

Keywords: Business Process Management, Root Cause Analysis, Machine Learning

1 Introduction

In a competitive, modern business era, being able to improve business processes is one of the most important ways to beat competition. When problems occur, the first instinctive response is to treat the symptoms as the problem and try to eliminate it. But this only corrects that isolated problem and allows similar problems to occur in the future. Root Cause Analysis (RCA) is a structured analytical problem-solving or failure-investigation technique that helps to determine and categorize the cause of any unfavorable event to eliminate the underlying root causes instead of only responding to the immediate superficial problem (12). The objective of RCA is to identify what event occurred, how it happened and why, which helps investigators to elaborate plans of action to correct the analysed processes and prevent future unfavorable events from happening (14). RCA can be used as a tool for identifying the cause of undesirable events as well as a tool for process improvement (12).

Traditional RCA approaches were based on manual process analysis through question schemes or other organizational analysis frameworks. But these approaches started becoming unfeasible with the amount of data being generated and stored, resulting in processes far more complex than before. In order to conduct RCA on complex processes and on high dimensional data, machine learning and data mining techniques started to gain popularity (17). These approaches has showed great results mainly due to the fact that machine learning algorithms need and thrive with large amount of data and are not hindered by the complexity of the processes being analysed.

In this paper a generic approach for RCA through machine learning is proposed. This paper is divided in five sections. Section 2 introduces related work on Root Cause Analysis and different approaches, Section 3 proposes a generic method for RCA with Machine Learning, Section 4 validates the proposed approach through a public loan application log from the 2017 BPI Challenge (4) and Section 5 presents the results of applying RCA to the aforementioned dataset.

2 Related Works

There are numerous approaches for performing RCA. Typically the Western approach of RCA is to use brainstorming to identify the most likely source for the undesired event with tools such as *Ford's 8D method* and *Six Sigma strategy*. The *A3 thinking* approach is purely based on facts but needs a baseline or defined standard of the problem to compare against. Other approaches such as the *fish bone diagram* and the *five why's technique* limit the investigation in a narrow spectre with a limited *why* question. Reid et al. even proposed a new method of using *5W + 1H procedure* where they ask what, who, when, where twice and how. They explicitly avoided using the why question because of the proved misinterpretation of this question. (12).

Although different RCA techniques revolve around different assumptions and possess different strengths, most of them have four defined steps(3; 12; 14):

1 **Data Recollection**: Data gathering mostly in the form of a process or an event log.
2 **Causal factor charting**: Sequence diagram for process precedence establishment of the recollected data and identification of major event contributors of the undesired incidence.
3 **Root cause identification**: Identification of root causes from the selected causal factors. Usually a Root Cause Map is used for structural reasoning purposes and for understanding why the causal factors occurred.
4 **Recommendation generation and implementation**: After the root cause identification of causal factors the investigators can elaborate recommendations and plans of action in order to eliminate the recurrence of the undesired incidence.

RCA techniques are usually applied in step two and three where data interpretation and problem identification are required. The analysis using traditional RCA methods can be subjective since they are highly depend on human interpretation and the methods vary vastly from each other (3; 12).

Traditional RCA models predominated the first Business Process Intelligence (BPI) competitions for process mining from 2011 to 2013, but since 2014 competitors have started using classification models for RCA analysis and data log interpretation(11). This trend can be easily understood by noticing the similarity that data mining and classification models have with the four defined steps of RCA, which makes implementation easier. They both rely heavily on recollected data and they both have some sort of data interpretation. On RCA's side it is in the form of causal factor identification and root cause identification. On the data mining and classification model's side this can be interpreted as a classification task of the undesired event or problem and the brainstorming part from the traditional RCA approach being replaced with feature selection and training models based on reality from recollected data. By using classification modeling, RCA is subject to less human interpretation and therefore less prone to biases. Machine learning algorithms also work better than the traditional RCA approaches when dealing with a complex business model with intertwined processes and a high volume of data since human analysis becomes highly hindered and unfeasible with high volumes of data (5; 6; 10; 15).

3 Method

In this paper a generic method guidance to combine RCA with machine learning is proposed. A graphical representation of the method is depicted in Figure 1.

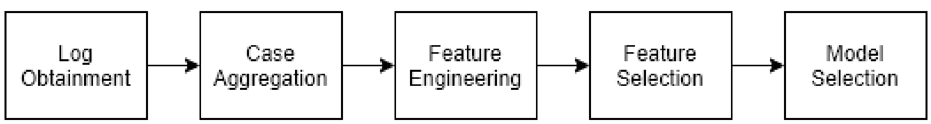

Fig. 1: Proposed Method

3.1 Event log obtainment

RCA is based on analyzing processes to find underlying problems or potential improvements, so it is necessary to obtain information on past events and processes that are typically recorded in event logs. For the logs to be useful, there are minimum requirements that need to be fulfilled. An event log must have a case identifier to distinguish between different cases. The log must also have an activity name to identify different activities that occur in one case. The last requirement is to have an activity sequence identifier so events can be retraced, for example event timestamps. Other attributes can also be present in event logs which can potentially help in identifying root causes (5; 15). Table 1 shows an excerpt of a sample event log that fulfills the minimum requirements.

3.2 Case Aggregation

The second proposed step is to convert the previously recollected event log into a case log where all events from the same case or process are grouped together. This allows the extraction of features from each case, resulting in a data frame with different cases in every row and characteristics or features in each column. Features added in this data frame are not limited to the explicit data from the log

Table 1: Excerpt of a generated event log

Case ID	Activity	Time Start	Time Complete	Application Type
Cust_1234	A_Create_App	01/01/2016 10:51:15	01/01/2016 10:51:17	New Credit
Cust_1234	A_Submitted	01/01/2016 10:51:17	01/01/2016 10:51:19	New Credit
Cust_5678	A_Create_App	01/01/2016 10:51:18	01/01/2016 10:51:20	Limit Raise

file but can also include insights and features deduced from the logs or obtained from other sources such as through social queries. This step is important because RCA is mostly conducted at a case level (15). An example of a data frame where case aggregation was applied is shown in Table 2.

Table 2: Excerpt of an aggregated event log

Case ID	Application Type	Total Case Duration	Requested Amount
Cust_1234	New Credit	1144676	2000
Cust_5678	Limit Raise	530018	10000

3.3 Feature Engineering

Feature Engineering is the process of deriving new features from features present in the data or through knowledge from domain experts. The objective is to improve model training accuracy and inference on unseen data with these added features (7; 19). Some important features to monitor/create are case duration, overtime of cases, handover rate and expert rate.

Case Duration can be derived from a case log as result in Section 3.2. Given a case log with start and end timestamps for each activity in the case, it is possible to derive the total duration of the case with the following formula. (15)

$$\text{Case Duration} = \text{Last Activity Time Complete - First Activity Time Start}$$

Overtime is a Boolean variable which can be derived directly from case duration. To create this feature, it is necessary to establish a threshold in order to assign 1 as overtime or 0 as on-time to this variable. The appropriate threshold for each log varies and must be analyzed through trial and error in conjunction with the classification step. Using the mean or median case duration is a good starting threshold. This variable can be used to identify delays in a process.

Handover rate is obtained by first calculating the handovers each case has. A handover is the amount of times work is handed from one employee to another. The handover rate is obtained by dividing the handover by the amount of activities in the case. This feature can be useful to quantify possible delays caused by excessive handovers. Table 3 provides an example of handovers in a case based on the flow diagram shown in Figure 2. A handover is registered when User 1 hands the case to User 2. When the same user does more than two activities in a row, no handover is registered. Table 4 shows the Handover Rate from this example.

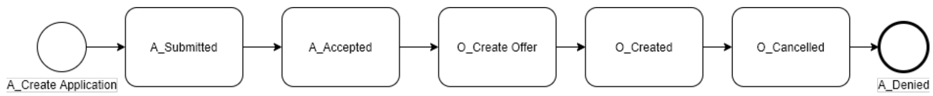

Fig. 2: Workflow Example

Table 3: Example of User-Activity representation in a case

Activity	Resource
A_Create Application	User 1
A_Submitted	User 2
A_Accepted	User 2
O_Create Offer	User 1
O_Created	User 1
O_Cancelled	User 1
A_Denied	User 3

Table 4: Handover Rate

#Handover	#Activities	Handover Rate
3	7	0.429

Expert Rate The involvement of experts can possibly have an impact on the outcome and duration (case performance). This theory can be tested by creating another feature called Expert Rate. The definition of an expert needs to be established when the information is not present. The expert rate compares two aspects: the number of total activities done by an employee and the involvement of the employee on each activity available. This feature is formally defined below.

Let X_{ij} be the sum of user i working on activity j, U_i be the sum of all activities done by user i and A_j the sum of activity j from all cases in the log. With this it is possible to calculate the percentage of activity j compared to all activities conducted for each employee with $I_{ij} = X_{ij}/U_i$. It is also possible to calculate the contribution of each employee on activity j with $R_{ij} = X_{ij}/A_j$.

To establish if an employee can be considered an expert in a specific activity, I_{ij} is compared to R_{ij}. If $I_{ij} > R_{ij}$, then user i is considered an expert for activity j. This expert information can be captured in a Boolean variable where 1 is considered an expert and 0 otherwise. A list of experts for each activity can be obtained through this process. An example of expert rate list calculation is presented on Table 6.

To identify the expert rate for each case, the number of experts present needs to be counted and divided by the total number of activities in each case. This process is illustrated in Table 7.

Data Transformation After creating all necessary features, it is important to analyze what approach to take for RCA. Since RCA is a tool for finding underlying or root issues in processes, we need to identify what issue or question RCA is going to be applied to, so it is important to understand the business process being analyzed. Two examples of important aspects to analyze in organizations are the outcome and duration of a process (12; 15).

Table 5: User-Activity Representation

Resource	Activity
User_1	A_Cancelled
User_1	A_Submitted
User_1	W_Shortened
User_1	A_Cancelled
User_2	A_Cancelled
User_3	A_Cancelled
User_3	A_Cancelled

Table 6: Expert List

Resource	Activity	X_ij	U_i	A_j	I_ij	R_ij	Expert
User_1	A_Cancelled	2	4	5	0.5	0.4	1

Table 7: Expert Rate

Case_ID	Activity	Expert	Expert Rate
Case_1234	12	7	0.58333
Case_5678	15	12	0.80000

Supervised classification machine learning models need predictor variables and response variables to train and classify the outcome, so it is necessary to convert RCA into a machine learning problem. This is done by using the previously identified question or issue as the response variable and all other attributes in a case (each row is a case) as predictor variables (10; 15). Reapplying data recollection is sometimes necessary after selecting the response variable to increase the number of predictor variables associated or relevant to the response variable.

Some machine learning algorithms have restrictions in the data type, only admitting numerical or categorical data, so it is important to convert to the necessary data type depending on classification model being used (8). One way to convert categorical data to numerical is to one-hot encode the data to create attribute fields for each level of the categorical value and assigning 1 or 0 depending on the existence of that variable in the case. Another way to encode these variables is by treating them as ordinal (ranked) data where a number is assigned to each category. But the second approach would bias data encoded with a higher number as better/worse which is intended in some classification problems.(18).

3.4 Feature selection

The previous step converted RCA problem into a classification problem. Now it is important to reduce computational complexity when training and increase model generalization performance in the validation dataset (7) as well as human interpretability (15) by applying feature selection.

It is important to distinguish feature selection from dimensionality reduction. Both reduce the number of attributes in a dataset but the later creates new combinations of features. The aim of using machine learning for RCA is to be able to interpret the classification process and obtain insight on root causes for problems. By creating new attributes from a combination of attributes, they become harder to interpret and insight into root causes is difficult to achieve. This is why dimensionality reduction algorithms are not common in RCA with machine learning. Two main categories of feature selection algorithms:

1 **Filter feature selection:** This type of feature selection ranks attributes with statistical methods such as correlation, mean, standard deviation, etc. Filter methods work well to identify and eliminate variable redundancy, which has a negative impact on model performance (7).
2 **Wrapper feature selection:** This method works by evaluating a machine learning model with a fixed subset of features and choosing the best subset of features. Some algorithms to search this feature subset are exhaustive search, best first, and greedy algorithm (2). This type of feature selection is important to reduce the number of attributes in the dataset, even further, improving human interpretability (2; 7; 15).

Model Selection There are numerous classification algorithms well suited for the task of classifying a particular response variable such as Decision Trees, Bayesian Networks, Artificial Neural Networks, Support Vector Machines and Logistic Regression (9). Machine Learning models with high interpretability are deemed White-Box models and models with low interpretability are deemed Black-Box models (13; 16).

The key element to take into consideration when deciding on the correct machine learning model for RCA is the interpretability of the model because RCA requires insights on the root causes of unfavorable events to eliminate them and prevent recurrences, therefore White-Box models are preferred(12).

Decision Trees are the most versatile and easy-to-interpret machine learning model since it allows categorical and numerical data as well as categorical data and the tree branching and creation is transparent. A problem with Decision Trees is that it tends to overfit all samples of the training data and performs poorly on unseen data. This is mitigated by limiting the branching depth of the tree and by using ensembles of Decision Trees (Random Forests) since it use different randomly selected features for each tree in the Forrest (9).

4 Method validation

The methodology for performing RCA with Machine Learning proposed in Section 3 is validated with the 2017 BPI Challenge dataset (4). All necessary steps and data transformations were performed with Python scripts (1).

Event log obtainment The 2017 BPI Challenge event log was used in this paper, which consists of a loan log from a financial dutch institution with which has 31509 cases. This log file contains all processes and events filed trough an online system in 2016 and their subsequent events until February 1st 2017 (4).

Case log creation In this step all events and processes in the log file are grouped into cases and stored in a data frame where rows represent cases and columns represent features in each case. Table 8 showcases a snippet of the Case Log.

Case ID	Application Type	Requested Amount	...
Application 652823628	New Credit	2000	...
Application 1691306052	Limit Raise	10000	...

Table 8: Case log data frame.

Feature Engineering Features presented in Section 3 *Case Duration*, *Overtime*, *Handover rate* and *Expert Rate*) are calculated and added to the case log. Conventional thresholds for *Overtime* cases were used, such as the mean and median duration of cases. Other thresholds were also experimented with. The results of applying these different thresholds and evaluating them with an entropy-based decision tree with 10-fold cross validation and maximum branch depth limit of 5 are shown on Table 9. The process of variable selection and model evaluation will be explained in detail in feature selection and model selection steps.

The minimum case duration in the log is 3 minutes and 21 seconds and the maximum case duration is approximately 170 days. The mean is approximately 22 days and the median is also approximately 19 days, standard deviation is approximately 12 days and 23 hours.

Table 9: Overtime threshold variation

Threshold	Accuracy	AUC	Number of features
10 days	82%	0.78	8
12 days	75%	0.78	2
19 days	76%	0.84	3
22 days	80%	0.88	5
24 days	83%	0.88	9
28 days	87%	0.91	8
33 days	91%	0.80	3

A trend can be seen in the results on Table 9. The farther the distance between the chosen threshold and the mean (22 days), the worse the accuracy and Area under the ROC curve (AUC) of the model becomes. The reason for this is that the log starts becoming imbalanced and therefore the AUC score is reduced. Another interesting result is that threshold lower than the median tend to have an accuracy and AUC lower than values that are higher than the mean. This can show that cases that take longer than the median might have more characteristics in common and therefore are better classified. Number of features also had the same tendency as AUC, the higher the threshold, the higher the number of features. In RCA it is beneficial to not have too many features for better interpretability. On further analysis of the graphical trees used for classifying, it could be seen that most of the high AUC models used the same features and produced similar trees. This shows that the mean and median are good blind pick threshold for Overtime variable. Ultimately the mean was chosen as the threshold in this paper because it was a balance between high accuracy, high AUC and low number of features. It is worth noting that dividing the log into 2 sections might be useful. This way intricacies in the data with higher and lower thresholds than the mean can be analyzed without the problem of creating an imbalanced classification problem.

The response variable was selected depending on what information was needed to be answered. In this case it is important to know the time in which cases are processed and the loan application outcome. This information can be obtained in the following variables: *Overtime* and *Application Status*. So these are the features selected as response variables.

Case ID	Application Type New Credit	Application Type Limit Raise	...
Application 652823628	1	0	...
Application 1691306052	0	1	...

Table 10: One-hot-encoded features data frame

After selecting the response and predictor variables, categorical features are one-hot-encoded. The process of one-hot encoding can be seen in the difference between Table 8 and Table 10. After one-hot encoding, the result is a classification-ready case data frame which can be used for machine learning or feature selection.

Feature Selection With the classification-ready data frame from the last section, it is important to conduct feature selection to reduce data dimensionality and thus improve model performance and interpretability. Filter feature selection is first applied to identify and remove feature redundancy. Spearman's Correlation was chosen for the correlation analysis because it is a non parametric correlation analysis and therefore requires no data distribution knowledge. Features that scored higher than 0.7 were removed. This reduced the amount of features from 30 to 25 features.

Wrapper feature selection is also applied to reduce dimensionality even further. For this step, an entropy based decision tree with a limited maximum depth of 5 and 10-fold cross validation is applied for Recursive feature elimination which evaluates the model with all the features and removes the worst entropy-ranking feature on every iteration until only one feature remains. Since two response variables were identified in the last section (Overtime and Application Status variables), feature selection analyses is applied to these two variables. For Overtime response variable, the optimal amount of features is 5 with an accuracy of 80% which can be seen in figure 3. For Application Status response variable the optimal number of features is also 5 with an accuracy of 95.5% which can be seen in figure 4. The selected features can be seen on table 11 for Overtime and table 12 for Application Status variables with their corresponding Information

Gain score. The blue shades in figures 3 and 4 represent the standard deviation of the 10-fold cross validated models.

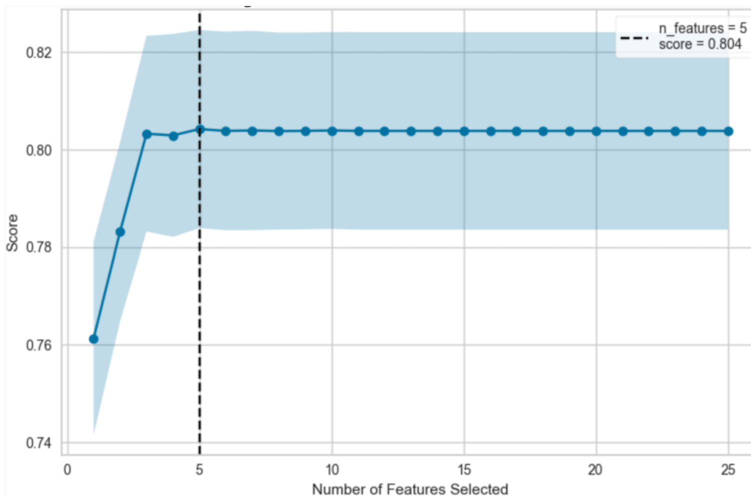

Fig. 3: Accuracy vs Number of features for Overtime response variable

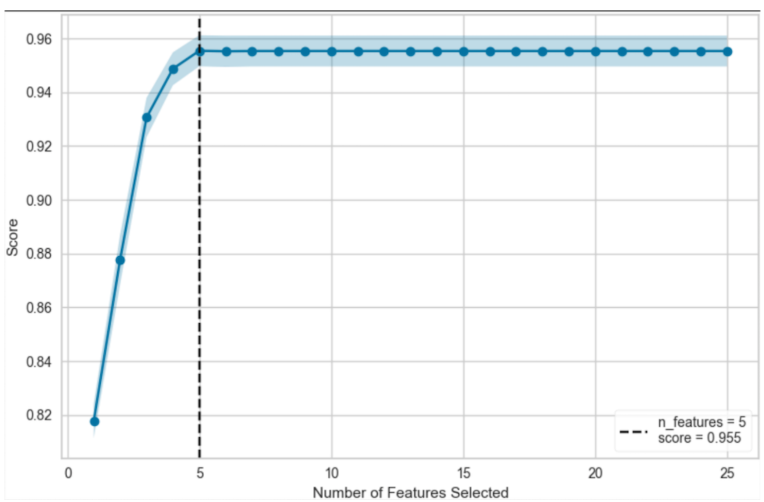

Fig. 4: Accuracy vs Number of features for Overtime response variable

Model Selection Using the output from the previous step where feature selection was conducted, it is now possible to select a machine learning model to use for RCA. Neural networks were tried first because of the notorious high accuracy they have for classification tasks. But they were discarded because of the lack of

Table 11: Best attribute subset for Overtime response variable

Attribute Name	Information Gain
Offered Amount	0.525
Expert Rate	0.256
Number of activities	0.191
Number of handovers	0.022
Loan goal car	0.005

Table 12: Best attribute subset for Application Status response variable

Attribute Name	Information Gain
Credit Score	0.587
Total case duration	0.168
Expert rate	0.128
Number of activities	0.072
Application type new credit	0.046

model interpretation which is necessary for RCA. Because of this decision trees are used with the same configuration as in feature selection: 5 level depth limit, entropy branch split criteria and 10-fold cross validation. This yields an accuracy of 96% for Application Status and 80% for Overtime.

RCA Analysis For a proper RCA analysis it is necessary to dive into the graphical decision trees used for each classification model to understand the intricacies of each classification.

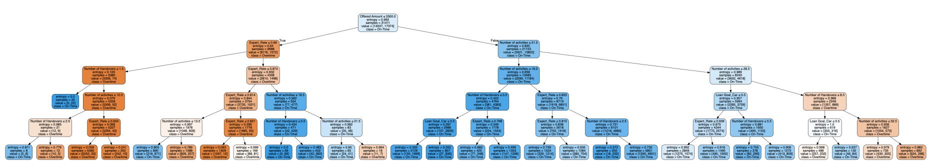

Fig. 5: Overtime Tree

Figure 5 is the complete graphical representation of the decision tree used for Overtime classification. Every node in the tree has the split criteria first, then the amount of entropy in that node, the number of total samples, values represent the amount of samples for each class and the class represents the classified class by the decision tree. Important information about overtime cases can be obtained in the root node seen in Figure 6. If Offered Amount is lees than or equal to 2500, then cases result in overtime. In another node in the tree, information about overtime cases regarding expert rate can be derived. If the Expert Rate for a case is greater than 0.874, cases result On-Time. This can be seen in Figure 7.

Figure 8 is the complete graphical representation of the decision tree used for Application Status classification.

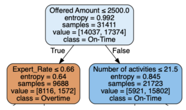

Fig. 6: Overtime root node

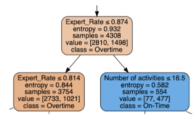

Fig. 7: Overtime node

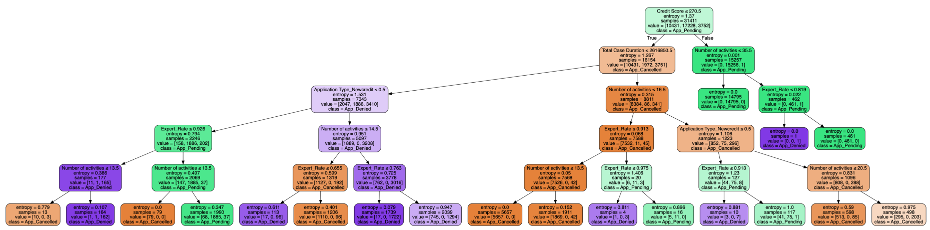

Fig. 8: Application Status Tree

Analyzing the root node for the Application Status tree in Figure 9, it is possible to derive information about the application status result for cases based on the credit score. Cases result in pending if the credit score is greater than 270. Cases resulted in cancelled by the client when they took longer than 30 days or 2616851 seconds.

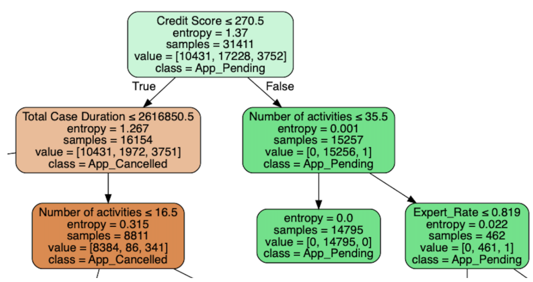

Fig. 9: Application Status Root Node

Another information derived from the Application Status decision tree node in Figure 10 is that if expert rate is less than or equal to 0.655, then cases result in cancelled. If Expert Rate is less than or equal to 0.763, then cases result in denied. In the other hand in Figure 11, if Expert Rate is greater than 0.926, then cases result in pending. This shows that cases with low Expert Rates tend to result in denied or cancelled cases.

The overall RCA findings in this data set is that the financial institution benefits greatly from cases with credit scores greater than 270, since they tend to result in ongoing cases. Cases should not take longer than 30 days or they are in risk of getting cancelled by the client. The duration of cases can be lowered

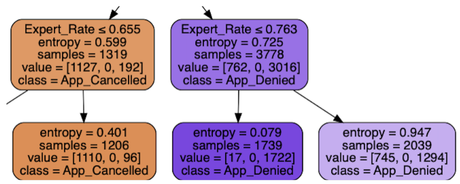

Fig. 10: Application Status Node

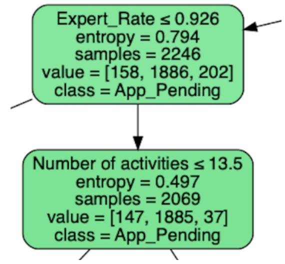

Fig. 11: Application Status Node

with high expert rates in cases, since it is seen that the presence of experts in cases reduce the overall case duration and increase the number of ongoing cases. Information on the number of handovers also needs to be taken in consideration. It is seen that handover rate has small influence on case duration or case Application Status, therefore it is in the financial institution's best interest to invest in employee training and specialization rather than to reduce the number of handover per case. Finally, to further improve in Case Duration and Application Outcome, it could be beneficial to divide the data into two sections and analyze different thresholds. This will give individual detailed information to each data division: long lasting cases and short lasting cases. This is only necessary if more insight into case duration is required.

5 Summary

The proposed generic method in this paper was proven to be effective in providing appropriate insights for RCA with machine learning when analyzing the graphical representations of the classification decision trees. This insights led to RCA recommendations and RCA root problem eradication. Since this method uses minimum characteristics present in log files, it is possible to apply it to any log file in any field of study. Using this method in comparison to traditional RCA methods, provides additional benefits such as the fact of being able to analyze large amounts of data which is important in the era of Big Data. This RCA solution also provides the analyst with some level of automation in finding connections and trends in the data. This method converts an RCA problem into a machine learning classification problem, so investing in data mining and classification techniques is highly rewarding when applying this method to other data sets.

Bibliography

[1] https://github.com/samik-real/machine-learning-in-bpm

[2] Aha, D.W., Bankert, R.L.: A comparative evaluation of sequential feature selection algorithms. In: Learning from data, pp. 199–206. Springer (1996)

[3] Chang, X., Terpenny, J., Koelling, P.: Reducing errors in the development, maintenance and utilisation of ontologies. International Journal of Computer Integrated Manufacturing 23(4), 341–352 (2010), https://doi.org/10.1080/09511921003642105

[4] van Dongen, B., Borchert, F.: Bpi challenge 2017. eindhoven university of technology. dataset. 2017

[5] Ferreira, D., Vasilyev, E.: Using logical decision trees to discover the cause of process delays from event logs. Computers in Industry 70 (03 2015)

[6] Gonzalez, J.M.N., Jimenez, J.A., Lopez, J.C.D., G., H.A.P.: Root cause analysis of network failures using machine learning and summarization techniques. IEEE Communications Magazine 55(9), 126–131 (Sep 2017)

[7] Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of machine learning research 3(Mar), 1157–1182 (2003)

[8] Hastie, T., Tibshirani, R., Friedman, J.: The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media (2009)

[9] Kotsiantis, S.B., Zaharakis, I., Pintelas, P.: Supervised machine learning: A review of classification techniques. Emerging artificial intelligence applications in computer engineering 160, 3–24 (2007)

[10] Liu, D., Zhao, Y., Xu, H., Sun, Y., Pei, D., Luo, J., Jing, X., Feng, M.: Opprentice: Towards practical and automatic anomaly detection through machine learning. In: Proceedings of the 2015 Internet Measurement Conference. pp. 211–224. ACM (2015)

[11] Lopes, I.F., Ferreira, D.R.: A survey of process mining competitions: the bpi challenges 2011–2018. In: 15th International Workshop on Business Process Intelligence (BPI'19), in conjunction with BPM. pp. 1–6 (2019)

[12] Reid, I., Smyth-Renshaw, J.: Exploring the fundamentals of root cause analysis: are we asking the right questions in defining the problem? Quality and Reliability Engineering International 28(5), 535–545 (2012)

[13] Ribeiro, M.T., Singh, S., Guestrin, C.: "why should i trust you?": Explaining the predictions of any classifier (2016)

[14] Rooney, J.J., Heuvel, L.N.V.: Root cause analysis for beginners. Quality progress 37(7), 45–56 (2004)

[15] Suriadi, S., Aalst, W., Ter, A.: Root cause analysis with enriched process logs. In: Lecture Notes in Business Information Processing. vol. 132, pp. 174–186 (01 2013)

[16] Turner, R.: A model explanation system. In: 2016 IEEE 26th International Workshop on Machine Learning for Signal Processing (MLSP). pp. 1–6. IEEE (2016)

[17] Van Der Aalst, W.M., Dustdar, S.: Process mining put into context. IEEE Internet Computing 16(1), 82–86 (2012)
[18] Zhang, K., Wang, Q., Chen, Z., Marsic, I., Kumar, V., Jiang, G., Zhang, J.: From categorical to numerical: Multiple transitive distance learning and embedding. In: Proceedings of the 2015 SIAM International Conference on Data Mining. pp. 46–54. SIAM (2015)
[19] Zhang, W., Du, T., Wang, J.: Deep learning over multi-field categorical data. In: European conference on information retrieval. pp. 45–57. Springer (2016)