Journal of Theoretical and Applied Information Technology
31st July 2021. Vol.99. No 14

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

THE APPLICATION OF MACHINE LEARNING APPROACH TO ADDRESS THE GPV BIAS ON POS TRANSACTION

[1]MUJIONO SADIKIN, [2]PURWANTO SK, [3]LUTHFIR RAHMAN BAGASKARA

[1]Faculty Member, Universitas Mercu Buana, Computer Science Faculty, Indonesia

[2]Faculty Member, Universitas Esa Unggul, Economic & Business Faculty, Indonesia

[3]BI Engineer, PT. Solusi Teknologi Niaga, Data Analytics Division, Indonesia

E-mail: [1]mujiono.sadikin@mercubuana.ac.id, [2]purwanto@esaunggul.ac.id. [3]Luthfirrahman04@gmail.com

ABSTRACT

Each transaction always produces junk data or bias data either due to errors or intentions. The junk data volume is always increase day by day, mainly in the using of public and free to use applications. Junk data is a disruption in every decision making which can cause the material or immaterial losses. This kind of problems are also occurring in the Qasir.id application, a POS application developed by PT. Solusi Teknologi Niaga for MSME entrepreneurs in Indonesia. In the company case, the junk data of POS transaction causes a poor quality of GPV (Gross Payment Value) information. The article presents the results of study in the POS transaction junk data handling. The junk data handling is performed by to validate three machine learning techniques and to deploy the best model in the company's Business Intelligence (BI) system. Based on the result of qualitative and quantitative evaluations, it is shown that the proposed approach provide a significant contribution to the company's decision-making process. The evaluation applied to the operational data sample reveals the accuracy score in the handling of junk data is 0.96 in precision, 0.73 in recall value, and the f1 score is 0.831. Whereas the qualitative evaluation based on users feed back of two-month operation indicates that users were greatly assisted in decision-making regarding the GPV.

Keywords: *Employee Appraisal, Additional Salary, Employee Performance, Decision Support System, FIS, Fuzzy Logic*

1. INTRODUCTION

Apart from its high complexity and large volume, one of the problems in the Big Data era at present time is junk data and data bias (*noise*). Junk data is the data contains anomalies so it is not standardized or inconsistent [1]–[4]. Some examples of anomalies are blur in images, non-standardized vocabulary or unnecessary words in text, or background noises of voice data. Junk data is the topic of concern and discussion material of the researchers owing to its effect on the data quality and lesss accuracy decision making. Some publications discuss this junk data are addressing data issues which do not meet the standards[5] and managing biased of image data [6]. In the second study, the image data is used to identify cleanliness of restrooms to help allocate cleaning service personnel. Furthermore, the junk data on credit card transactions is the subject of this paper [7].

Junk data or noise can arise from all data recording systems. The quantity of junk data is getting higher for open data recording transactions such as e-commerce or other open data recording systems. The presence of junk data disrupts the various analyses and reporting needs of an organization [8], a company's business, employee performance reports, and even gross value of a start-up company

This junk data issues also occur at PT Solusi Teknologi Niaga (Qasir.id), a start-up company who engage in developing point-of-sale applications by name" Qasir". The point-of-sales (POS) application developed to assist the MSMEs (Micro Small and Medium Enterprises) in recording their online and offline transactions, managing products, and monitoring transaction reports without paying for application services. Due to the fact that this point-of-sales application is free, it leads to the consequence that lots of "trial" transactions done by merchants which affect on one of the performance indicators at PT Solusi Teknologi Niaga i.e GPV (Gross Payment Value). GPV is the merchant transaction value recorded in the system, but it is not counted as company profits.

The GPV indicator is calculated based on user behavior in using the POS application. There are three categories of user behavior, namely: *test*, *real active*, *stop user/slipped away*. However, since not all categories of behavior are used in the calculation of GPV, separation of user behavior is needed to prevent the calculation of GPV from being biased.

This paper presents the results of research aiming for managing and analyzing junk data to separate the three user behaviors using the Machine Learning approach. Several open-source tools from python were used in data pre-processing, modeling, and model implementation [9]. The initial stage of the research is comparing of the performance of several algorithms in order to get the best one. The results of this research have been implementing for 2 months on a cloud server by using the open-source ETL Pipeline from *apache*. Based on the qualitative evaluation carried out by gathering user opinions, it is concluded that the implementation of the research results are feasible to be used for the company operations.

Research related to the management of "junk data" has been widely conducted by other researchers. The management of "junk data" is done by different methods and dataset. This section points out several studies related to managing the "junk data".

The first study is the management of the deodorant dataset and the address conducted by K Hima Prasad, et al [5]. In this study, the authors investigated the framework to standardize sentences inputted by user. The first step is to investigate the dataset by identifying patterns in each row. Next, data segmentation and correction of the wrong words were carried out. The RDR Framework is used to help correct words automatically in new data. In this RDR Framework, researchers applied rule-based method that resulted in quite good performance with an average precision of 0.6 and recall of 0.6. Shiyang Xuan, et al [7] discussed research to deal with the issue of fraudulent use of credit cards. The study is started by data labeling which performed by using predetermined parameters based on the usage history of user's credit card. The researchers applied 2 different Random Forest Algorithms. RF1 provided the precision and recall of 90.27% and 67.89%, whereas RF2 provided the precision and recall of 89.46% and 95.27%.

The study on noise in images published by Lahiru Jayasinghe, et al [6] used a hygiene image dataset from restrooms with the PCA (Principal Component Analysis) method for pre-processing it and the CNN model as its algorithm. To manage the noise, color augmentation method is deployed. Then the PCA method is applied to the image resulting from the color augmentation process. At the modeling stage the CNN algorithm is applied. This CNN based model is used to predict dirty, average, or clean rest-room categories. The only weakness in this study is on having images from 1 type of restrooms.

The next study about data noise is conducted by Etaiwi et al [10] which applied the a dataset published by Myle Ott et al [11]. In the dataset contains many spam reviews and fake reviews which had an impact on online marketplace behavior. In the study, the authors used bag-of-words and words counts methods to detect spam reviews. Four algorithms were compared, namely naïve-bayes, random forests, decision trees and support vector machines. For accuracy evaluation purposes, the accuracy, precision and recall were used. There were two stages of evaluation. The first is the result of feature selection of bag-of-words and words counts. In the evaluation of words counts feature selection, the best accuracy, precision, and recall is from the naïve-bayes algorithm. In the evaluation of the bag-of-words feature selection, the best accuracy and recall were shown by naïve-bayes with 87.305% and 92.632% respectively, while the best precision is indicated by random forests with 64.784%.

To the best of the author knowledge, there are no studies performed in dealing with data disruption of GPV-related POS transactions, while the real issue occurs in the field. Therefore, in this study authors propose and implement the research results for the purpose. The research is conducted through several stages, namely dataset labeling based on predetermined parameters [7], [12], comparison of several algorithms [10], the implementation of the best model on the cloud server, and the quantitative and qualitative evaluation on the model implementation results

The rest of the paper is organized as follows. Section two discusses the related study regarding the discussed topic. In the next section, section tree, it is presented the material and method used in the study. The data processing and computational mechanism are also presented in the section two. The experiment results, its deployment evaluation and the discussion are discussed in the third section. The last section presents the conclusion and the future work related to the subject

2. RESEARCH METHOD

This research applied experimental methods, system implementation, and surveys to get feedback from users. The experiments were conducted to get the best model which is implemented in the BI system. In the experimental stage the validation of three classification algorithms, namely: Random Forests, Decision Trees, and Logistic Regression, are carried out. The best algorithm of the validation results is then implemented in the BI system. After the using of the BI for two months, a survey is conducted to obtain feedback on the use of the BI system.

2.1. Classification

The classification in this research are used to sort out the types of transactions consisting of active user transactions, unsustainable transactions, and merely experimental transactions. Classification is a technique in data mining or machine learning used to classify dataset based on label or target class. Hence, algorithms or methods for solving classification problems are categorized as supervised learning. The purpose of supervised learning is in which label or target attribute acting as a 'teacher' or 'supervisor' who guides the machine learning process in order to achieve a certain level of accuracy or precision [13]. Some algorithms or methods that can be used to solve classification problems such as Random Forest, C.45 or better known as Decision Tree, Logistic Regression, Naïve Bayes, Deep Learning and othersD'Urso et al, as presented in the publication [14], study the MCDM (Multi Criteria Decision Making) in fuzzy logic to support decision making that able to accommodate many complex criteria. The author proposes the fuzzy logic hierarchy method to overcome some issues associated with the uncertainty and the vagueness of specific decisions in very complex and multi-criteria frameworks. Based on the experiment results, author conclude that the method can be improved to get the optimum solution.

2.2. Decision Tree

Decision Tree is one of the most popular classification methods as it is easy to interpret by humans. Decision Tree is a classification method that applies a tree structure representation, each node representing an attribute, a branch representing the value of an attribute, and a leaf representing a class or target. Regardless its easy interpretation, decision tree has a lack of efficiency in analysis and in its level of accuracy. [13], [15]. The decision tree construction is based on the selection of dataset attributes used as a node at each stage of the tree development. The node is choose based on the information gain computed by formulas (1) and (2)

$$Gain(S,A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} \quad (1)$$

Remarks:
S : The Sets of cases
A : Attribute
n : The number of Partitions attribute A
$|S_i|$: Number of cases in the I partitions
$|S|$: Number of cases on S

Which, to calculate entropy is as follows:

$$Entropy = \sum_{i=1}^{n} -pi * \log_2 pi \quad (1)$$

Remarks:
S : The Sets of cases
A : Feature
n : The number of Partitions S
pi : The proportions of S_i against S

2.3. Random Forest

Random Forest is a classification method as the Decision Tree. The basic concept of this method is to create a collection of trees by randomly selecting attributes. In developing and analysing the tree, random forest consumes less time because the tree created will have only a few attributes. In cases, the accuracy of this method is better compared to the Decision Tree method as the classification results do not only depend on one tree but many trees [16], [17]. Another interesting of fuzzy variant applied in the human resources management area is ANFIS (Adaptive Neural Fuzzy Inference System) which is proposed by Krichevky et al [18]. In supporting the decision making on employee candidate selection, the author proposed a multi layers decision making system. The multi layers configuration is combination of NN and fuzzy logic. The intermediate output of this architectures is the regression equation which connects the candidate quality with his/her characteristics. Whereas in their publication [19], authors present the study result of the using Random Forest classifier to classify text dataset in fishery domain. By tuning its parameters, the best accuracy performance result achieved is 0.95.

The class prediction is performed based on those tree votes which is computed as the formula (3).

$$p(c \mid v) = \frac{1}{T}\sum_{t=1}^{T} p_t(c \mid v) \quad (2)$$

Remarks:
p(c|v) : Forest Class
T : Size of Forest
$p_t(c|v)$: each tree leaf yields the posterior
t : number of trees

2.4. Logistic Regression

Logistic Regression is a classification algorithm used for probability prediction by comparing data on logit functions of logistic curves. Unlike the Linear Regression which produces a target output in the form of continuous data, the Logistic Regression output produced is categorical data computed by the formula (4) [20].

$$P(Y) = \frac{e^{b_0+b_1x_1+b_2x_2+...+b_nx_n}}{1+e^{b_0+b_1x_1+b_2x_2+...+b_nx_n}}$$

Remarks:
P : probability of Y occuring
e : natural logarithm base
b_0 : interception at y-axis
b_1 : line gradient
b_n : regression coefficient of X_n
X_1 : predictor variable

2.5. Dataset

The dataset used in the study is merchant transaction data through the Qasir point-of-sales application. The dataset is obtained by querying a table in the production database, then exporting it to a CSV file. Each instant data consisted of 34 attributes, with 47.506 instant data collected. All attributes has a numeric type as in table 1. "day1" to "day31" attributes were the number of transactions carried out by the merchant on the first day to the 31st day of the same month. "Month" described the current month and "Year" is the current year.

Table 1. Example of Dataset

merchant_id	month	year	day1	…	day31
213124	8	2019	100	…	12
192920	9	2019	200	…	1
828293	10	2019	0	…	0

2.6. Research Stages

The research stages carried out were divided into six steps. The first one is dataset labeling based on the predetermined parameters [12] using the calculation of the number of transactions from 'day 1' to 'day 31'. The second step is the pre-processing stage applying the Feature Selection method, by eliminating some unnecessary attributes [21]. The third stage is the modeling performed by to compare those three different algorithms, namely Random Forest, Decision Tree, and Logistic Regression. The fourth stage is to validate those three algorithms. The algorithm which provided the best results is then implemented on the cloud server. The fifth stage is the implementation of the best model on the cloud server with the help of the open-source ETL Pipeline from *apache*. The last stage is the evaluation of model having been implemented for first of two months. In this final stage, quantitative and qualitative evaluations were used. Illustration of the above stages is presented as in figure 1.

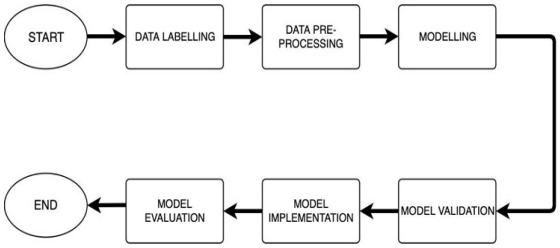

Figure 1. Experiment Stages

2.6.1. Data Labelling

Data Labeling in this study is carried out using the python package namely *pandas* and *numpy* to transform data based on predetermined parameters [12]. There are three categories of labels defined such as Real Active User, Stop User/Slipped Away, and Testing User. The first thing to do is to calculate the number of values from column day1 to day31 valued 0. If the total value of 0 obtained is less than or equal to 20 then it is categorized as active user, and other than that it is categorized as User Testing. Then the data labeled as active user is separated into real active user and stop user/slipped away by calculating the number of values of 0 from column day24 to day31. If the total value of 0 is more than 7 then it is categorized as Stop User/Slipped Away, while the rest are Real Active Users.

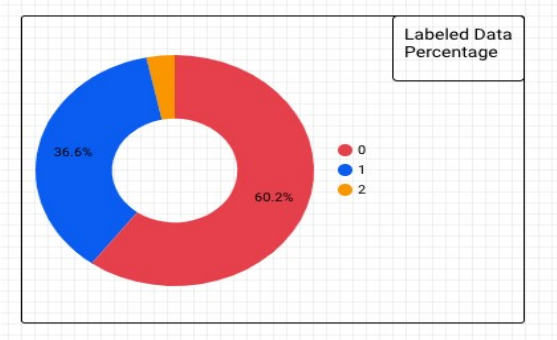

Figure 2. Data labelling composition

The result of dataset labelliing is depicted as figure 2. The meanings of label 0, 1 and 2 are Testing User, Real Active User, and also User Stop respectively. The amount of dataset instant of each label consqutively are label 0 28,577 merchants (60.2%), label 17,370 merchants (36.6%), and the rest are total merchants of label 2

2.6.2. Data Pre-Processing

Data pre-processing is carried out to improve the quality of classification results. This process is performed by removing unnecessary columns or changing a value or object in the data instant [22]. Pre-processing conducted in this research is Feature Selection to select the certain attributes used for modeling purpose. The attributes removed from the dataset are merchant_id, month and year to adjust the research focus on the patterns exisiting on the day1 to day31 attributes as presented as figure 3.

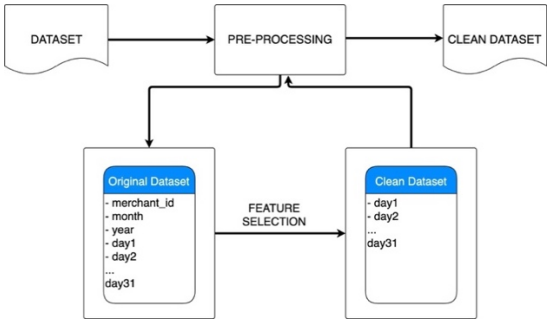

Figure 3. Feature Selection Process

2.6.3. Modelling

In this research modeling is done by testing three classification algorithms, namely: Random Forst, Decision Tree, and Logistic Regression. These three algorithms were tested using two test scenarios based on the separation of training data and testing data. The first data separation scenario is random splitting which sorted training data from testing data with the composition of training data: testing data are 70:30, 80:20, and 90:10 respectively. The second scenario of sorting data sets used k-fold cross with k-fold values of 5, 10, 15, and 20 respectively.

2.6.4. Model Validation

The next stage is model validation which is carried out to measure the performance of the three classification algorithms tested. The accuracy performance of the model is validated using precision, recall, and f1 score parameters. Validation schemes were to ensure that the model really performs well in predicting new data. The calculation of the three parameters are based on the confusion matrix [23]–[25]. Based on the confusion matrix, the performance indicators are compute. The performance indicators are TP, FP, TN, and FN as shown in table 2.

Table 2. Confusion Matrix Definition

Name	Definition
TP (True Positive)	The number of positive data considered true
FP (False Positive)	The number of positive data considered false
TN (True Negative)	The number of negative data considered false
FN (False Negative)	The number of negatiev data considered true

a. Precision

Precision is the classification ratio of positive data considered true to number of positive data considered true and false [10], [24].

$$precision = \frac{TP}{TP + FP} \quad (4)$$

b. Recall

Recall is the number of classification ratio of positive data considered true to the number of positive data considered true and negative data considered false [10], [24].

$$recall = \frac{TP}{TP + FN} \quad (5)$$

c. F1-Score

F1-Score is the result of 2 times precision and recall then divided by precision plus recall [26].

$$F1 = \frac{2*(\mathrm{Pr}ecision*\mathrm{Re}call)}{(\mathrm{Pr}ecision+\mathrm{Re}call)} \quad (6)$$

2.6.5. Model Evaluation

The best model obtained from the modeling stage is selected to be implemented in the BI system. Evaluation of the model is performed after the BI system has been operating in two months. Model evaluation is carried out by using two methods, namely Quantitative Evaluation and Qualitative Evaluation. Quantitative evaluation is conducted by comparing the results of BI predictions and the data labeling done manually. Transaction data used for this evaluation is collected from transaction data in October 2019. Qualitative evaluation is carried out using survey methods to find out the feedback from respondents who use BI in the daily operation.

3. RESULTS AND DISCUSSION

3.1. Random Splitting Validation Results

The validation results of the random splitting scenario experiments are presented as in table 3, table 4, and table 5 as well. The results of the trial of all random splitting schemes show that the Random Forest achieved the best f1-score values compared to the other two algorithms with the scores are 0.893, 0.877, and 0.877 respectively. In terms of processing time, RF performance is the least good as it required the longest time for the three splitting schemes.

Table 3. Random Splitting 70:30

Classifier	70:30			
	Precision	Recall	F1	Time(s)
DT	0.876	0.861	0.868	0.25
RF	0.955	0.84	**0.893**	2.97
LR	0.809	0.876	0.841	2.46

Table 4. Random Splitting 80:20

Classifier	70:30			
	Precision	Recall	F1	Time(s)
DT	0.875	0.868	0.871	0.31
RF	0.942	0.822	**0.877**	3.55
LR	0.868	0.798	0.831	3.15

Table 5 Random Splitting 90:10

Classifier	70:30			
	Precision	Recall	F1	Time(s)
DT	0.875	0.861	0.867	0.34
RF	0.946	0.818	**0.877**	4
LR	0.867	0.794	0.828	3.79

3.2. Cross Validation Results

The results of *Cross Validation* experiment scenarios with various *K-Fold* values is presented as tables 6, 7, 8 and 9. The performance results of the cross-validation experiments scenario confirm that RF provided the best results of the f1-score value. For each k-fold, 5, 10 and 15, Random Forest achieved f1-scores values of 0.836, 0.858, and 0.860 respectively which were higher than the 2 algorithms. Although for k-fold 20, Random Forest and Decision Tree had the same f1-score of 0.866, but on average for all three schemes, RF remained the best.

Table 6. K-Fold 5 Results

Classifier	K-Fold 5			
	Precision	Recall	F1	Time(s)
DT	0.879	0.818	0.832	1.33
RF	0.916	0.796	**0.836**	13.95
LR	0.878	0.794	0.804	10.33

Table 7. K-Fold 10 Results

Classifier	K-Fold 10			
	Precision	Recall	F1	Time(s)
DT	0.898	0.840	0.856	3.10
RF	0.932	0.822	**0.858**	31.31
LR	0.901	0.817	0.831	23.87

Table 8. K-Fold 15 Results

Classifier	K-Fold 15			
	Precision	Recall	F1	Time(s)
DT	0.902	0.841	0.858	6.97
RF	0.933	0.823	**0.860**	56.33
LR	0.907	0.820	0.836	40.01

Table 9. K-Fold 20 Results

Classifier	K-Fold 20			
	Precision	Recall	F1	Time(s)
DT	0.910	0.846	**0.866**	11.21
RF	0.940	0.827	**0.866**	81.85
LR	0.914	0.826	0.844	58.06

3.3. Model Implementation

Based on the experiments results, it is cobcluded that RF on average gave the best results for the f-score accuracy performance. This results indicated that RF is more suitable for the characteristics of POS transaction data. Therefore, the Random Forest model is selected to be implemented in a Business Intelligence (BI) application. The model implementation for BI in the operational environment ise developed by using the open-source ETL Pipeline from Apache called Airflow. Airflow is a ETL Pipeline with a batching process that applies the Python programming language [27]. For the purpose of server deployment, a compute engine from the Google Cloud Platform is used [28]. Data Warehouse used is Cloud SQL based on

Journal of Theoretical and Applied Information Technology
31st July 2021. Vol.99. No 14

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

MySQL [29]. The first process undertaken is to deploy the model into Airflow. Airflow withdrew data from datasource and loaded the data into a model that had been deployed. The model is then loaded back into the data warehouse. The implementation scheme can be seen in figure 4.

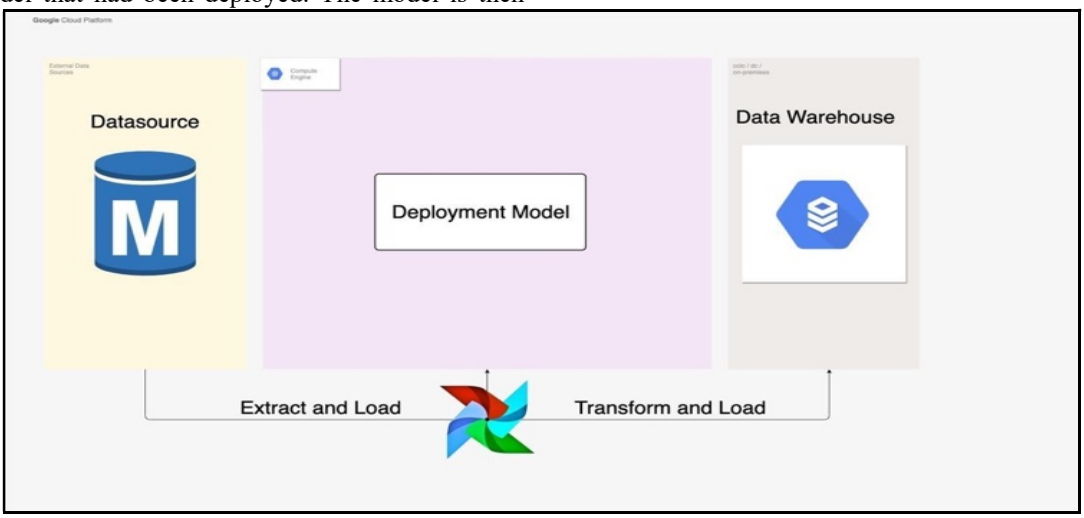

Figure 4. Model Implementation Schema

At the stage of importing the real data into the model, a Feature Selection process is carried out. Feature selection aims to select only the features needed in running the model, while attributes that are not used by the model are still used for visualization. The scheme of sorting and combining attributes in a real dataset is presented as in figure 5, whereas table 10 presents dataset output results.

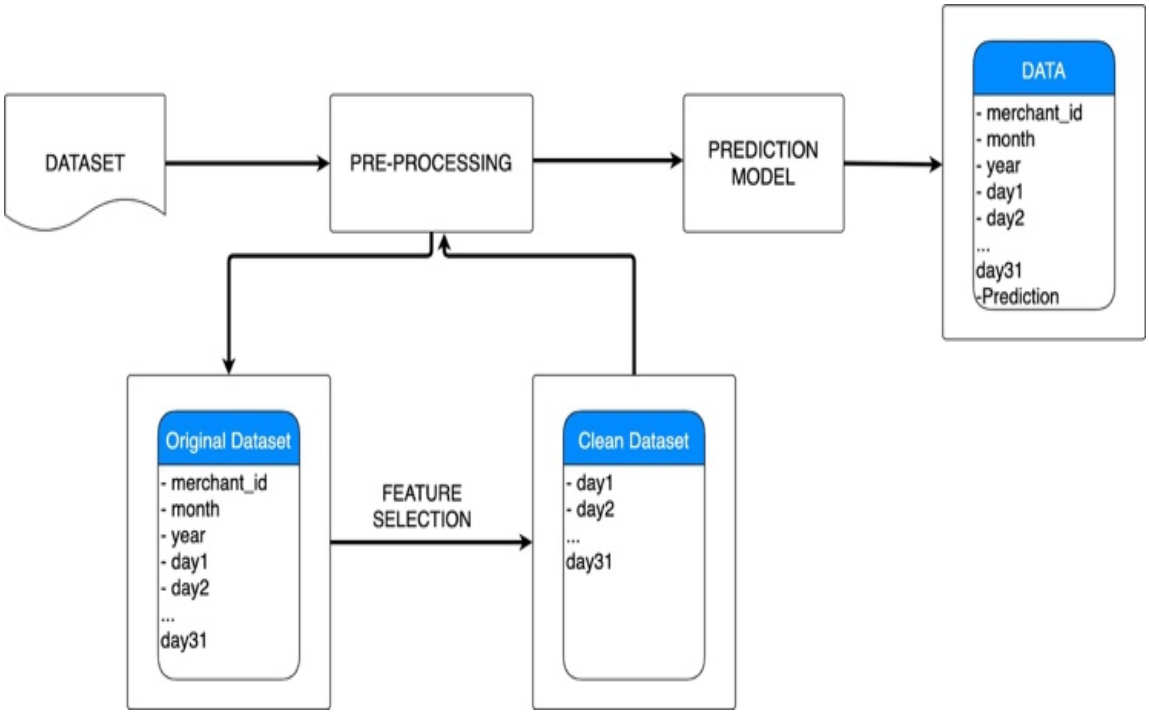

Figure 5. Feature Selection on Deployment Model

Table 10. Output Data Example

merchant_id	month	year	day1	…	day31	prediction
213124	8	2019	100	…	1	2
192920	9	2019	200	…	2	1
828293	10	2019	0	…	10	0

3.4. Deployment Model Evaluation

Based on the model evaluation at the experimental stage, the Random Forest algorithm is selected to be implemented in a BI application with a model on the cloud server. A portion of the BI application interface is presented as figure 6. The x-axis of the graph in Fig 6 represents the month of the transaction, while the y-axis represents the number of transactions. In each month period, the BI application presents three types of transactions based on the predicted model implemented, such as: Real Active User with a blue line, User Stop with a yellow line and User Testing with a red line.

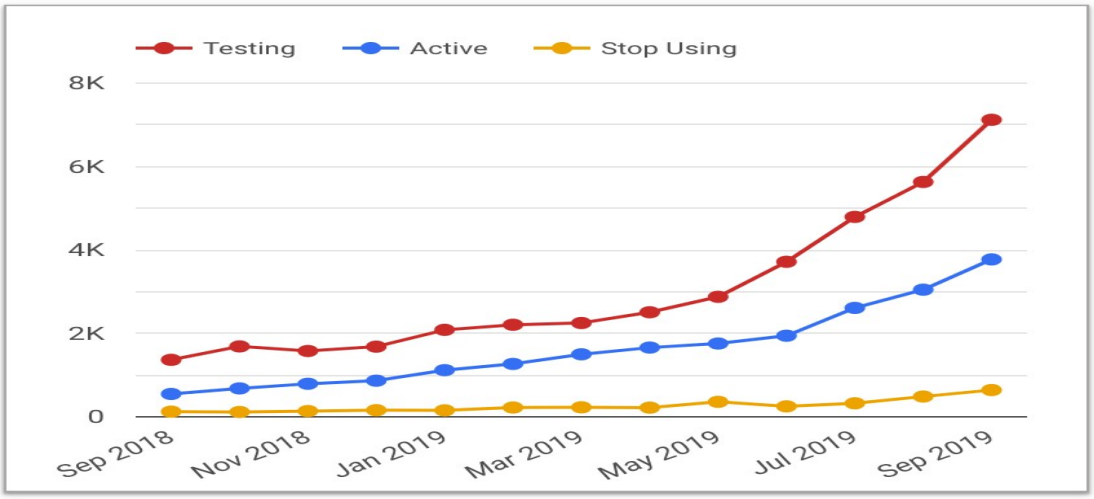

Figure 6. Merchants Status Graph

After the random forest model has been implementing for 2 months, quantitative and qualitative evaluations were carried out. Quantitative evaluation is conducted by comparing the results of prediction and labeling performed manually using transaction data in October. This evaluation is done by taking about 10000 real data analyzed by the model. The prediction results provided by the model of six thousand of data were verified manually. The model performance based on the results of manual verification is presented in table 11. Quantitative Evaluation applied the Confusion Matrix basis for Precision and Recall calculations [30], [31]. From the Confusion Matrix in table 5, label 0 is the testing transaction, 1 is the active transaction, and label 2 is the user stopped transaction. The model correctly predict 6570 of class 0 out of 7416, and those predicted as class 1 and class 2 were 342 and 504, respectively. The model also correctly predicted 3232 of class 1 and 32 of class 0. For class 2 the model predicted 100% accurately the 216 out of 216. The results of precision and recall computation is presented aas table 12, the precision is still at 0.95, but the recall is down by around 10% to 0.734.

Table 11. Confusion Matrix Result

Label	Confusion Matrix		
	0	1	2
0	**6570**	342	504
1	32	**3237**	0
2	0	0	**216**

Table 12. Quantitive Evaluation Result

Precision	Recall	F1-Score
0.958	**0.734**	**0.831**

To validate the operation performance of the BI system supported by the seletected random forest model we performed a qualitative analysis. The analysis is carried out by gathering feedback from users regarding the performance of the implemented model. Feedback is obtained by distributing questionnaires to BI users based on the

implemented model. A summary of the questionnaire results is presented in figure 7. In terms of BI user representation, respondents represented the data custodian, product manager, top management, marketing staff, business staff, financial, and customer satisfaction divisions. The parameters evaluation included the intensity of the use of BI per day, the frequency of days using the BI application per week, and the accuracy of the information presented by BI. Most users come from the Data Division and the Product or Development Division. Before using the Machine Learning based model for Dashboard Analysis of BI applications, the use of BI in one day is on the average of 63.15 minutes and in a week, it is used at most for 2 days. After the model is implemented, the use of BI increased to 120.4 minutes/day. The frequency of the use of BI per week also increased to 5 days selected by 8 people, followed by 4 days, 3 days, and 2 days, respectively by 6, 5, and 1 person. Finally, at the level of accuracy prior to Machine Learning, many selected "rather accurate", but after Machine Learning, many selected "Accurate".

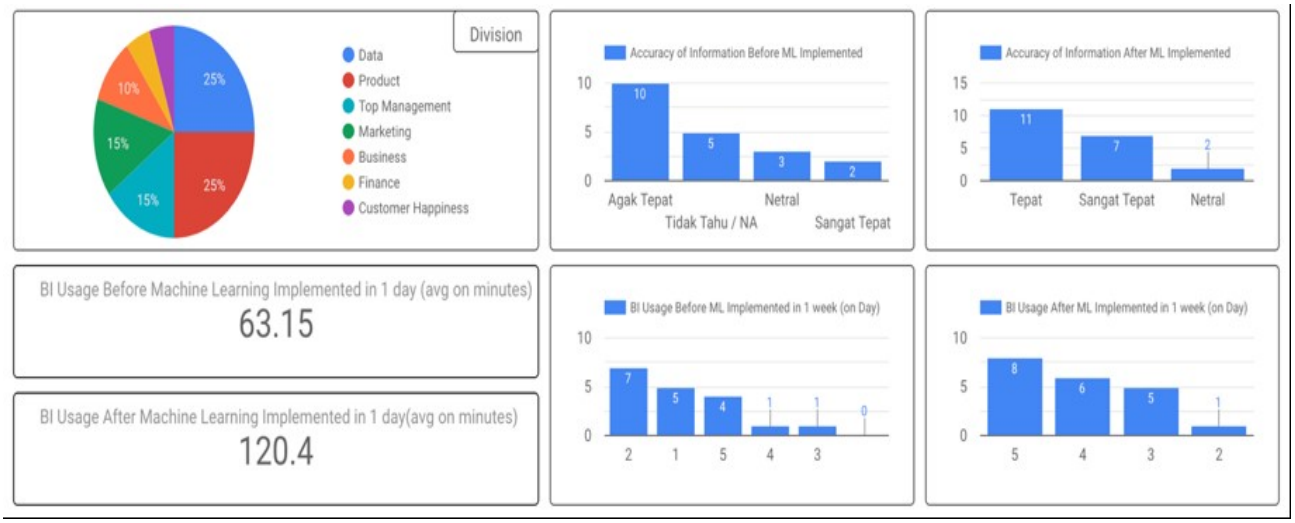

Figure 7. Questionnaire Result

3.5. Prior and This Work Analysis

GPV bias is the essential issue has to be adressed in order to the decision-making, especially in terms of investment, can be carried out more precisely. However, the solution to this problem in the computational point of view as long as our knowledge has not been touched at all, even only in the experimental level. In this study we not only conducted an experiment to find the best technique to solve this issue, but also implemented the best Random Forest model experimental results in a real operational environment. Based on observations of the use of models in the BI system over three months periods, the proposed solution is proven able to assist the better decision making. The beneficial of our proposed solution is convinced by the feedback collected from respondents consisting of various user roles.

4. CONCLUSION

The research contribution presented in this paper is the creation of a Machine Learning-based BI application that helps improve the quality of decision making in organizations. In this research, the issue of junk data/bias in POS transaction is addressed as the issue hindered the decision-making owing to the fact that GPV information became biased. GPV bias is caused by the presence of noise in the form of trial transactions. Managing this issue is done by selecting the appropriate machine learning technique as the core engine of a company's BI application. From the results of the model development experiments, it is found that the Random Forest algorithm owned the best performance. This random forest-based algorithm model is then implemented in a BI application. Qualitative and quantitative evaluation on the implementation and use of BI applications showed that the research results provided significant benefits in improving the quality of decision making. This is indicated by user feedback pointing out a positive increase in terms of frequency of use, intensity of use, and speed of decision making.

In the next research, we plan to further explore the data generated by POS transactions. Further prospects of data exploration include: the purpose of product/service recommendations, prediction of quantity and quality of transactions at merchants, and analysis of merchant behavior at the company level.

ACKNOWLEDGMENTS

The authors would like to thank PT. Solusi Teknologi Niaga (Qasir.id) for permitting the use of data and information on business process knowledge provided. The writer also appreciates the willingness of Mrs. Aulia Permata Sari and Mr. Heri Husaeri Achsan to be the reviewer of this article.

REFRENCES:

[1] S. Kumar and M. Singh, 'A novel clustering technique for efficient clustering of big data in Hadoop Ecosystem', *Big Data Min. Anal.*, vol. 2, no. 4, pp. 240–247, 2019.

[2] S. Krishnan *et al.*, 'SampleClean: Fast and Reliable Analytics on Dirty Data', *Bull. IEEE Comput. Soc. Tech. Comm. Data Eng.*, pp. 59–75, 2015.

[3] S. Juddoo, 'Overview of data quality challenges in the context of Big Data', *2015 Int. Conf. Comput. Commun. Secur. ICCCS 2015*, 2016.

[4] M. Zhou, Y. Wang, A. K. Srivastava, Y. Wu, and P. Banerjee, 'Ensemble-Based Algorithm for Synchrophasor Data Anomaly Detection', *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2979–2988, 2019.

[5] K. H. Prasad, T. A. Faruquie, S. Joshi, S. Chaturvedi, L. V. Subramaniam, and M. Mohania, 'Data cleansing techniques for large enterprise datasets', *Proc. - 2011 Annu. SRII Glob. Conf. SRII 2011*, pp. 135–144, 2011.

[6] L. Jayasinghe, N. Wijerathne, C. Yuen, and M. Zhang, 'Feature Learning and Analysis for Cleanliness Classification in Restrooms', *IEEE Access*, vol. 7, pp. 14871–14882, 2019.

[7] S. Xuan, G. Liu, Z. Li, L. Zheng, S. Wang, and C. Jiang, 'Random forest for credit card fraud detection', *ICNSC 2018 - 15th IEEE Int. Conf. Networking, Sens. Control*, pp. 1–6, 2018.

[8] S. Salloum, J. Z. Huang, and Y. He, 'Exploring and cleaning big data with random sample data blocks', *J. Big Data*, vol. 6, no. 1, p. 45, 2019.

[9] L. Thurner *et al.*, 'Pandapower - An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems', *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, 2018.

[10] W. Etaiwi and A. Awajan, 'The Effects of Features Selection Methods on Spam Review Detection Performance', *Proc. - 2017 Int. Conf. New Trends Comput. Sci. ICTCS 2017*, vol. 2018-Janua, no. 2, pp. 116–120, 2018.

[11] M. Ott, C. Cardie, and J. T. Hancock, 'Negative deceptive opinion spam', *NAACL HLT 2013 - 2013 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Proc. Main Conf.*, no. June, pp. 497–501, 2013.

[12] L. Zheng, G. Liu, C. Yan, and C. Jiang, 'Transaction fraud detection based on total order relation and behavior diversity', *IEEE Trans. Comput. Soc. Syst.*, vol. 5, no. 3, pp. 796–806, 2018.

[13] M. Sadikin, F. Afiandi, and F. Alfiandi, 'Comparative Study of Classification Method on Customer Candidate Data to Predict its Potential Risk', *Int. J. Electr. Comput. Eng.*, vol. 8, no. 6, 2018.

[14] M. G. D'Urso and D. Masi, 'Multi-Criteria Decision-Making Methods and Their Applications for Human Resources', in *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2015, vol. XL-6/W1, no. June, pp. 31–37.

[15] N. Quadrianto and Z. Ghahramani, 'A very simple safe-Bayesian random forest', *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 6, pp. 1297–1303, 2015.

[16] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, 'Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection', *IEEE Access*, vol. 6, pp. 33789–33795, 2018.

[17] A. Criminisi, J. Shotton, and E. Konukoglu, 'Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning', *Found. Trends*

Comput. Graph. Vis., vol. 7, no. 2–3, pp. 81–227, 2011.

[18] M. L. Krichevsky, J. Martunova, and V. Sirotkin, 'Neuro-fuzzy recruitment system', *Espacios*, vol. 38, no. 62, p. 15, 2017.

[19] D. Ramayanti and U. Salamah, 'Text Classification on Dataset of Marine and Fisheries Sciences Domain using Random Forest Classifier', *Int. J. Comput. Tech.*, vol. 5, no. 5, pp. 1–7, 2018.

[20] H. Khurshid and M. F. Khan, 'Segmentation and classification using logistic regression in remote sensing imagery', *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, vol. 8, no. 1, pp. 224–232, 2015.

[21] H. Liu, X. Li, and S. Zhang, 'Learning Instance Correlation Functions for Multilabel Classification', *IEEE Trans. Cybern.*, vol. 47, no. 2, pp. 499–510, 2017.

[22] B. Vinzamuri, Y. Li, and C. K. Reddy, 'Pre-processing censored survival data using inverse covariance matrix based calibration', *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 10, pp. 2111–2124, 2017.

[23] J. L. García-Balboa, M. V. Alba-Fernández, F. J. Ariza-López, and J. Rodríguez-Avi, 'Homogeneity test for confusion matrices: A method and an example', *Int. Geosci. Remote Sens. Symp.*, vol. 2018-July, pp. 1203–1205, 2018.

[24] M. Ohsaki, P. Wang, K. Matsuda, S. Katagiri, H. Watanabe, and A. Ralescu, 'Confusion-matrix-based kernel logistic regression for imbalanced data classification', *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 9, pp. 1806–1819, 2017.

[25] A. Aksjonov, P. Nedoma, V. Vodovozov, E. Petlenkov, and M. Herrmann, 'Detection and Evaluation of Driver Distraction Using Machine Learning and Fuzzy Logic', *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2048–2059, 2019.

[26] Z. C. Lipton, C. Elkan, and B. Naryanaswamy, 'Thresholding Classifiers to Maximize F1 Score'.

[27] Sally, 'the Apache Software Foundation Announces Apache® Airflow™ as a Top-Level Project', 2019. .

[28] G. Cloud, 'Compute Engine', 2018. [Online]. Available: https://cloud.google.com/compute/. [Accessed: 09-Oct-2018].

[29] 'Cloud SQL for MySQL documentation'. [Online]. Available: https://cloud.google.com/sql/docs/mysql/features. [Accessed: 14-Oct-2019].

[30] A. Tharwat, 'Classification assessment methods', *Applied Computing and Informatics*, 2018.

[31] B. H. Shekar and G. Dagnew, 'A Multi-Classifier Approach on L1-Regulated Features of Microarray Cancer Data', *2018 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2018*, pp. 1515–1522, 2018.