E-Commerce Customer Churn Prediction Using Machine Learning Approaches

Bagaskara Putra Wibowo[1]*, Lili Ayu Wulandhari*[2]
[1,2]*Computer Science Department, Binus Graduate Program–Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia*
E-mail: bagaskara.wibowo001@binus.ac.id[1]*, lili.wulandhari@binus.ac.id*[2]

Abstract

E-commerce businesses face the challenge of retaining customers in the era of rapid digital expansion. Customer churn prediction becomes essential for strategic decision-making by offering insights into potential revenue loss and customer loyalty. One of the problem in customer churn prediction comes from the presence of outliers in the data. This research delves into seeing the effects on churn prediction f1-score by incorporating a combination of techniques including outlier detection via k-means clustering and DBSCAN, as well as employing XGBoost and Catboost as classifiers. Results indicate that using Catboost gives a better performance of 96% F1-Score for e-commerce customer churn dataset with outliers, and removing outliers does not result in an increase in performance.

Keywords: *Customer Churn, Outlier Removal, DBSCAN, K-Means Clustering, XGBoost, Catboost*

1. Introduction

E-commerce refers to the use of the Internet and other electronic media for commercial purposes such as dealing goods and services from businesses to consumers. Rivaling companies are required to innovate and incorporate several strategies so they can maintain a presence within the market and fulfill the ever-changing requirements of the competitive scenario [1]. One of such way is through predicting customer churn as maintaining loyal customers will have an effect on the business's revenue and profitability [2]. Data from the e-commerce platform can be used to do customer churn prediction through machine learning approaches.

Challenges in e-commerce customer churn prediction includes high data dimensionality as well as imbalance data where data of either the churn or non-churn class has a significantly higher number than the other. Another challenge is the presence of outliers in the data where data that differ significantly from the other data exist within the dataset. The presence of outlier data within a customer behavior dataset may indicate unique and exceptional instances where customers acted outside their usual expected pattern (i.e. a one-time large purchase) and can result in an increase in complexity and decrease in performances of the prediction model [3]. Handling these outlier data may in turn increase the performance of the machine learning model and results in a better classification.

Several methods can be used to do outlier handling for the proposed intention. The first one is through the use of classifier models that are robust to outliers such as gradient boosting. Ahmad et al. [4] and Mittal [5] have compared the use of gradient boosting algorithms for customer churn prediction and both research results indicate that gradient boosting outperforms the other algorithms, with a resulting accuracy of 93% and 96% respectively. Lubis et al. [6] similarly conducted a comparison between logistic regression and gradient boosting for predicting customer churn and gained a higher performance from the latter in both train and test data. Another way to handle the outliers is by removing them from the dataset. One of the ways to achieve this is by the use of clustering such as k-means clustering and DBSCAN. Clustering algorithms can be used to detect anomalies within the dataset as has been done for problems including hospital claims [7] and credit card transactions [8]. Outlier removal using k-means clustering has been done by Wu et al., [9] on a diabetes dataset and the resulting diabetes prediction has a high accuracy of 95.42%. Alirezaei et al., [10] conducted similar method on multiple datasets and the resulting accuracies range from 94% to 100%. These high results suggest that this method of outlier removal is excellent for classification purposes and may yield similarly high performance when applied to an e-commerce dataset as done in this research.

This research paper attempts to test several combinations of outlier removal and classification methods for the purpose of customer churn prediction. XGBoost and Catboost will be used as the classification model, while k-means clustering and DBSCAN are used for outlier removal purpose. A total of six combinations will be tested namely: XGBoost with no outlier removal, Catboost with no outlier removal, XGBoost with k-means clustering for outlier removal, Catboost with k-means clustering for outlier removal, XGBoost with DBSCAN for outlier removal, and Catboost with DBSCAN for outlier removal. The performance of each combination will be compared and the one with the best predictive outcome is concluded as the best customer churn prediction method.

2. Research Methodology

The problem to be resolved in this research is customer churn classification, in which customers of an e-commerce platform are classified into either churn (leaving the e-commerce platform) or non-churn (loyal to the platform). This is done by using the available customer data and feeding them into some gradient boosting classifiers. Initial exploratory data analysis shows that the dataset contains outliers on each of its features, and based on the referenced works mentioned in the first section, removal of outliers can yield a better performance. Two clustering methods are applied into the data first before classification for outlier removal.

There are two clustering methods proposed for the purpose of outlier removal from the customer churn dataset. The first is through k-means clustering. This method first clusters the data into a number of clusters containing a mix of churn and non-churn data. Any data that belong to the minority class in each cluster are treated as outliers and are removed. The second is through DBSCAN. DBSCAN detects outliers by analyzing the density and connectivity of data points. It labels data points as core points, border points, or outliers based on their proximity to other data points. Outliers in the dataset are identified as data points that fall outside the dense regions formed by the majority of data points to be removed.

Three experiments are done in the research. The first experiment acts as a baseline where data are immediately fed into the XGBoost and Catboost classifiers as is without outlier removal. The second and third experiment add k-means clustering and DBSCAN respectively for outlier removal before the training phase. The F1-score of each experiments are obtained anc compared. The effect of varying the percentage of outliers being removed is also done in the second experiment with k-means clustering. The research design is given in Figure 1.

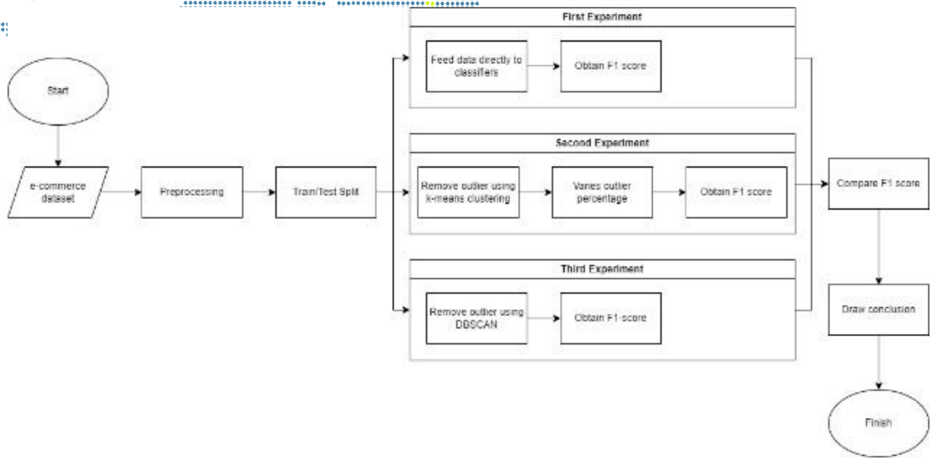

Figure 1. Research Steps

2.1. Dataset

The dataset that will be used in this study is the "Ecommerce Customer Churn Analysis and Prediction" dataset. The dataset is available in the public domain and is free to use, and is relatively new so it has yet to be used often on past research. This dataset was obtained by downloading from the Machine Learning and Data Science community site, Kaggle.

The dataset contains data belonging to a certain leading online E-Commerce company. The dataset contains a total of 5630 different data with various data types. Each data in the dataset has a label 0 which indicates non-churn, and label 1 which indicates churn. There are 20 different attributes in the dataset contains a mix of categorical and numerical values as well as multiple missing values on several features. The differing data types will be handled by using the appropriate encoding methods for each feature, and the missing value will be filled in by using either the mean or the median of each feature based on the presence of outliers in each feature.

2.2. Data Preprocessing

Several preprocessing are done to the data including the imputation of missing values by using each features' median, feature encoding to transform all categorical features into numerical representation, as well as standard scaling. All aforementioned preprocessing methods are done on all three experiments.

The first step after the dataset is entered into the system is the stage of correcting missing values. The dataset has several missing values in several of its attributes, including Tenure,WarehouseToHome, OrderAmountHikeFromlastYear, CouponUsed, OrderCount, DaySinceLastOrder, and HourSpendOnApp. These missing values are presented with the NaN value. These missing values can be replaced by using either of the attribute's mean or median, depending on the presence of outliers within each attribute. Mean will be used if there are no outliers, while the reverse holds true for median. Figure 2 shows the flowchart of missing values handling.

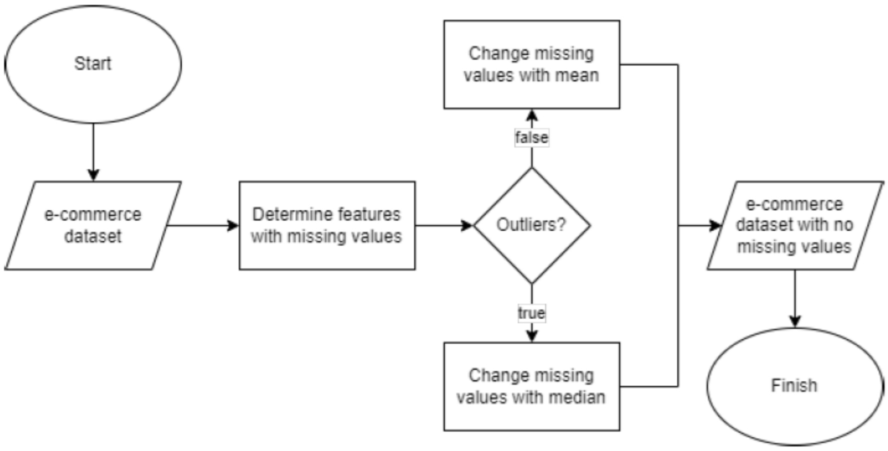
Start
e-commerce dataset
Determine features with missing values
Outliers?
false
Change missing values with mean
true
Change missing values with median
e-commerce dataset with no missing values
Finish

Figure 2. Missing Values Handling Flowchart

Feature encoding is then done on all the categorical features: Gender, MaritalStatus, PreferredLoginDevice, PreferredPaymentMode, and PreferedOrderCat. Two types encoding are done depending on whether the attribute has binary categorical values or multiple categorical values. Label encoding is done on the former while one-hot encoding is done for the latter. Out of all the categorical values within the dataset, only the "Gender" attribute has binary values, while the remaining four have multiple categories. Figure 3 shows the flowchart of lshows the flowchart of one-hot encoding.

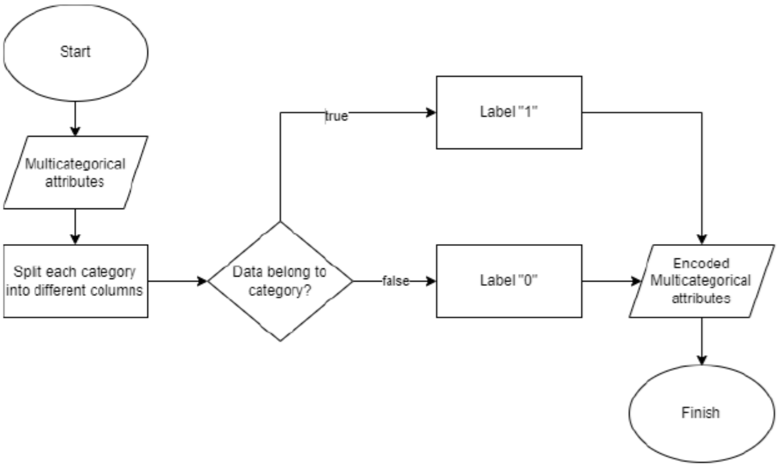
Start
Multicategorical attributes
Split each category into different columns
Data belong to category?
true
Label "1"
false
Label "0"
Encoded Multicategorical attributes
Finish

Figure 3. One-Hot Encoding Flowchart

2.3. First Experiment: Baseline

The first experiment done in this research is used as a benchmark for comparing the affect of outlier handling on predicting customer churn on the e-commerce dataset. The 5630 data first undergoes missing value handling by the replacing the missing values in each of the 20 attributes with its respective median. The next step is to do feature encoding (label and one-hot) on the categorical attributes of the dataset so that all the data are numerical. Standard scaling is then done on the resulting dataset before being split into 80% training set and 20% testing set. The training set is used to train an XGBoost and Catboost classifiers and the F1-score of both are obtained. Figure 4 shows the flowchart of the first experiment.

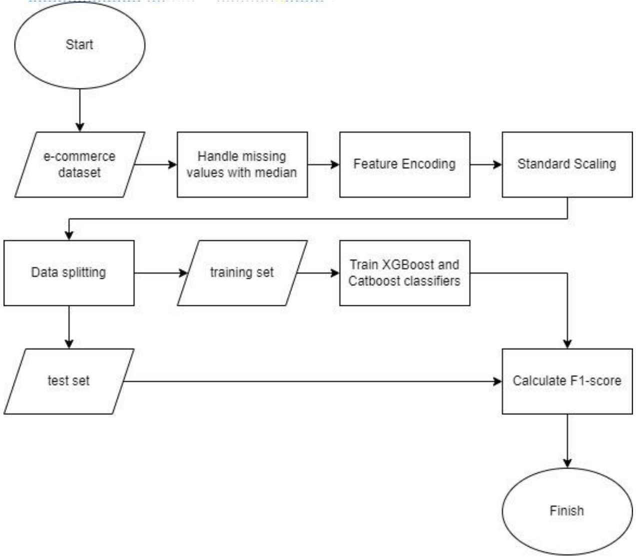

Figure 4. First Experiment Flowchart

2.4. Second Experiment: Outlier Removal with K-Means Clustering

The second experiment removes the outliers within the dataset using k-means clustering. The flow of the second experiment is similar to the first baseline experiment in which the dataset undergoes missing value handling, feature encoding, and standard scaling. The outlier removal process is added just after the train-test splitting. The imbalanced condition of the data necessitates the use of sampling to balance the data first in order for the k-means clustering outlier removal method to apply, since otherwise the non-churn class will always be the minority in all clusters and therefore be considered outliers. SMOTE is used to balance the data before the outlier removal process. The outlier removal result is then fed into the XGBoost and Catboost classifiers.

For the second experiment specifically, an additional step before the training process is done by varying the percentage of outliers being removed from the dataset. The outlier removal percentage ranges from 100% where all outliers are completely removed, to only 10% of outliers being removed. This is done to see the effect of varying presence of outliers to the classification model, and whether there is a possible benefit in keeping some of the outliers in the dataset rather than removing them completely. Figure 5 shows the flowchart of the second experiment.

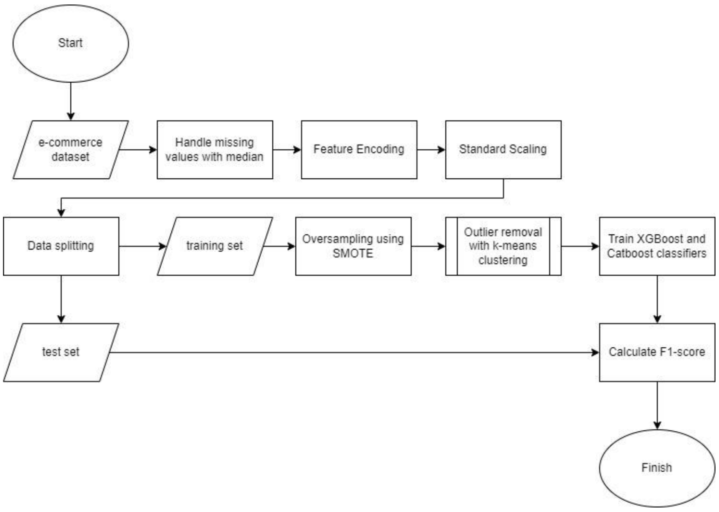

Figure 5. Second Experiment Flowchart

The outlier removal process involves clustering the data into several clusters. The number of churn and non-churn data within each cluster is then counted. Whichever class is a minority is each cluster is then treated as outliers and is completely removed. Figure 6 shows the flowchart of the outlier removal process using k-means clustering.

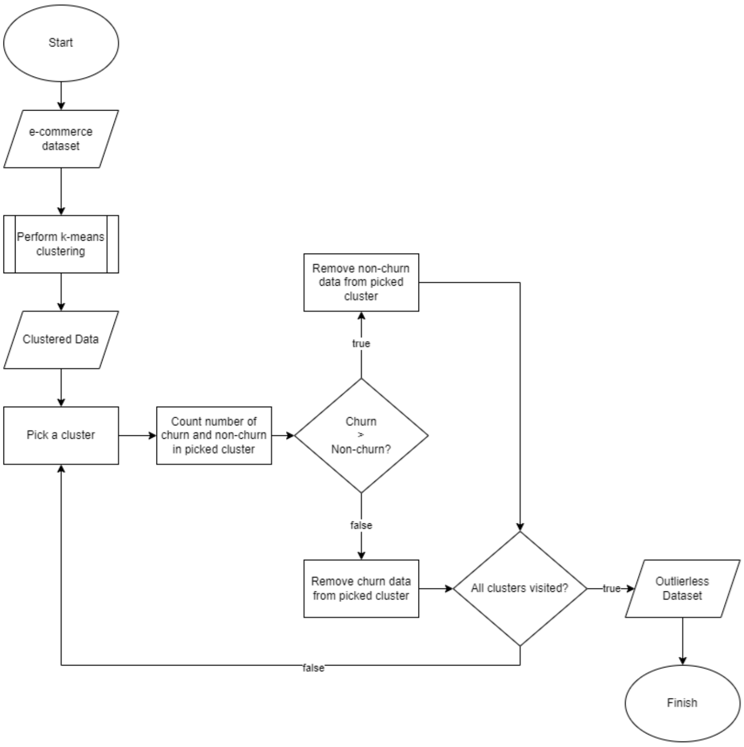
Start
e-commerce dataset
Perform k-means clustering
Clustered Data
Pick a cluster
Count number of churn and non-churn in picked cluster
Churn > Non-churn?
true
Remove non-churn data from picked cluster
false
Remove churn data from picked cluster
All clusters visited?
true
Outlierless Dataset
false
Finish

Figure 6. Outlier Removal with K-Means Clustering

2.5. Third Experiment: Outlier Removal with DBSCAN

The third experiment is similar to the second experiment in that it eliminates the outliers within the dataset, but through the utilization of DBSCAN instead of k-means clustering. It also uses XGBoost and Catboost algorithms as the classifiers as with the previous two experiments. DBSCAN clusters the data based on the distance between the individual data (epsilon) and a cluster is formed once a minimum number of data exists within the aforementioned distance. Data that are not clustered due to being too far away are then labeled as outliers and then removed. The flow of experiment is similar to the second experiment, where outlier removal is done after data splitting. The main difference is that no data balancing is done on the dataset since DBSCAN can still apply even with the imbalanced data. The F1-score of the first, second, and third experiments are obtained and then compared which are presented in the following third section.

3. Result and Discussion

Table 1 shows the result of the first experiment where the data undergoes preprocessing and then are immediately fed into into the classifiers. Several classifiers in addition to the XGBoost and Catboost that were proposed are also used in order to see the performance of different classification models on the dataset.

Table 1. First Experiment Result

No	Model	Highest Train F1-Score (%)	Highest Test F1-Score (%)
1.	XGBoost	100	95.2
2.	Catboost	100	96

The results of the first experiment show that the selected methods, Catboost in particular, outperforms XGBoost when it comes to F1-score, with a performance score of 96% over 95.2%. The second and third experiments will use XGBoost and Catboost classifiers only, with their respective results compared to the XGBoost and Catboost F1-score of the first experiment as seen in the table

Table 2 shows the result of the second experiment where the data undergoes preprocessing and then undergoes outlier removal using K-Means Clustering. The second experiment was conducted ten times separately by altering the percentage of the outlier data being removed ranging from 100% outliers removed to 10% outliers removed. This was done to see the effect of the amount of removed outliers on the performance of the model.

Table 2. Second Experiment Result

No	Outlier Removed (%)	Highest Train F1-Score (%)	Highest Test F1-Score (%)	Algorithm
1.	100	100	47.9	XGBoost
2.	90	100	70.5	Catboost
3.	80	100	76.8	Catboost
4.	70	100	84	Catboost
5.	60	100	85.2	Catboost
6.	50	100	86.7	Catboost
7.	40	100	87.2	Catboost
8.	30	100	90.7	Catboost
9.	20	100	90	Catboost
10.	10	100	92.6	Catboost

A trend that can be seen from the results of the second experiment is that when k-means clustering is used as the outlier removal method, removing more and more outliers from within the dataset results in a decrease in F1-score, with the highest test score obtained when only 10% of the outliers removed. Catboost outperforms XGBoost in most of the experiments with the latter only having better performances when the percentage of outliers removed is high. The result may indicate that the classifiers have less tendency to overfit if some outliers are kept within the dataset, thus allowing the model to learn from the presence of outliers.

Table 3 shows the result of the third experiment where the data undergoes preprocessing and then undergoes outlier removal using DBSCAN. Unlike the second experiment, the removal of 100% outlier as detected by DBSCAN does not result in overfitting and therefore, the removed outlier percentages are not varied.

Table 3. Third Experiment Result

No	Model	Highest Train F1-Score (%)	Highest Test F1- Score (%)
1.	XGBoost	100	94
2.	Catboost	100	95

Using DBSCAN as the outlier removal method, Catboost and XGBoost exhibited similar F1-score on the test set with 94% and 95% respectively. Catboost shows a slightly higher performance in comparison, similar to the results of the first and second experiment. Table 4 shows the comparison of all three experiment performances.

Table 4. Final F1-Score Comparison of All Three Experiments

No	Experiment	Method	Highest F1-Score (%)	Algorithm
1.	First	Baseline	96	Catboost
2.	Second	K-Means Clustering	92.6	Catboost
3.	Third	DBSCAN	95	Catboost

Based on the obtained results, it can be seen that the baseline first experiment where no outliers are removed has the highest test F1-score compared to the rest. The third experiment with outlier removal using DBSCAN has a very minute decrease with a test F1-score of 95%. The highest performance of the second experiment has the lowest score with 92.6%. Catboost generally outperforms XGBoost in all three experiments.

The obtained results indicates that the classifiers being fed data with no outliers being removed (the first experiment) performs better than when it is fed outlierless data (the second and third experiment). The experiments done in the second experiment where the percentage of outliers being removed is varied also show a similar trend, where a lower percentage of outliers being removed results in a higher F1-score, the highest being at 10% outlier removed. This suggests that gradient-boosting algorithms such as XGBoost and Catboost, exhibited strong performance in the presence of outliers, with Catboost being superior for this particular e-commerce dataset. This finding is consistent with previous studies that have highlighted the robustness of ensemble methods, including gradient boosting, to outliers in the data. XGBoost and CatBoost, are capable of handling noisy data effectively by building decision trees that adapt to the distribution of the data, effectively downweighting the influence of outliers. The ensemble nature of these algorithms further enhances their resilience to noise in the data. As a result, the models trained on the raw data may capture valuable information from the outliers, leading to improved predictive performance.

The experiments involving outlier removal using k-means and DBSCAN, on the other hand, demonstrated inferior predictive performance. One possible explanation for this outcome is that the outlier detection techniques themselves may not have accurately identified the true outliers in the data. Both k-means and DBSCAN rely on distance-based clustering, and their performance is contingent on the appropriate choice of hyperparameters, such as the number of clusters or the minimum number of data points in a cluster. If these hyperparameters are not chosen optimally, non-outliers could be misclassified as outliers or vice versa, leading to a loss of information that might be critical for prediction.

The third experiment that used DBSCAN for outlier removal performed better than the second experiment, which used k-means. The F1-score of the third experiment only has a very small difference compared to the baseline experiment. A possible benefit can be obtained by considering that the DBSCAN-based outlier removal may have an advantage in other metrics, such as training time. The use of DBSCAN can shorten the time taken for the models to train as some of the data are removed, while also not experiencing a significant drop in performance.

4. Conclusion

This research have investigated the use of XGBoost and Catboost for customer churn prediction, exploring three distinct experiments to gauge their impact on predictive performance. The experiments involved the direct input of data into the classifiers, outlier removal using K-means, and outlier removal using DBSCAN. The results of these experiments have provided valuable insights into the role of outliers in the context of customer churn prediction.

The most compelling finding of this study was that the first experiment, where the data was fed directly into the classifiers without outlier removal, yielded the best predictive performance. This outcome challenges the common belief that a clean dataset, achieved through the removal of outliers, is essential for achieving accurate predictive models. Our results suggest that the presence of outliers may in fact be beneficial for gradient boosting models such as XGBoost and Catboost tasked with customer churn prediction. For the purpose of customer churn prediction using this particular dataset, it can be concluded that using Catboost with no outlier removal results in the best predictive performance of 96% F1-Score.

Future avenues that can be done in the field of customer churn prediction in addition or in place of removing outliers include other preprocessing methods, such as the use of feature engineering to better capture the information contained within the outliers. These engineered features could enhance the model’s ability to leverage outlier information effectively. Other improvements can also be done to the algorithms being used for the outlier removal and classification to obtain a better result.

Acknowledgments

The authors would like to thank to Bina Nusantara University for supporting this research.

References

[1] A. Thakur, “Trends and Analysis of E-Commerce Market: A Global Perspective”, International Journal of Applied Marketing and Management vol. 6, **(2021)**, pp. 11–22.

[2] P. Mathai, "Customer Churn Prediction:A Survey", International Journal of Advanced Research in Computer Science, **(2020)**.

[3] A. Amin, A. Adnan, and S. Anwar, “An adaptive learning approach for customer churn prediction in the telecommunication industry using evolutionary computation and Naïve Bayes”, Applied Soft Computing, vol. 137, **(2023)**, p. 110103.

[4] A. K. Ahmad, A. Jafar, and K. Aljoumaa, ‘Customer churn prediction in telecom using machine learning in big data platform’, Journal of Big Data, vol. 6, no. 1, **(2019)**, p. 28.

[5] M. K. Mithal, "Customer Churn Analysis in Telecom Using Machine Learning Techniques", Masters thesis, School of Computing, National College of Ireland, Dublin, Ireland, **(2023)**.

[6] A. Lubis, S. Prayudani, J. Polmed, O. Nugroho, Y. Lase, and M. Lubis, "Comparison of Model in Predicting Customer Churn Based on Users' habits on E-Commerce",5th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), **(2022)**, pp. 300–305.

[7] H. Prakosa and N. Rokhman, ‘Anomaly Detection in Hospital Claims Using K-Means and Linear Regression’, IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 15, no. 4, **(2021)**, pp. 391–402.

[8] H. Wu, S. Yang, Z. Huang, J. He, and X. Wang, "Type 2 diabetes mellitus prediction model based on data mining", Informatics in Medicine Unlocked, vol. 10, **(2018)**, pp. 100–107.

[9] S. Sugidamayatno and D. Lelono, 'Outlier Detection Credit Card Transactions Using Local Outlier Factor Algorithm (LOF)', IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 13, no. 4, **(2019)**, pp. 409–420.

[10] M. Alirezaei, S. Niaki, and A. Niaki, "A bi-objective hybrid optimization algorithm to reduce noise and data dimension in diabetes diagnosis using support vector machines", Expert Systems with Applications, **(2019)**.