3rd International Conference on Information Science and Technology Innovation (ICoSTEC)
July 27, 2024, Yogyakarta, Indonesia

Comparative Analysis of Naive Bayes and K-Nearest Neighbors Algorithms for Customer Churn Prediction: A Kaggle Dataset Case Study

Anggraeni Xena Paradita [1], Nathifa Agustiana [2], Asriana [3], Putri Utami Rukmana [4],
Putri Nelsa [5], Muharman Lubis [4]

School of Industrial Engineering, Telkom University
Bandung, Indonesia
[1]anggraenixena@student.telkomuniversity.ac.id
[2]nathifaa16@gmail.com
[3]asrianayaya@student.telkomuniversity.ac.id
[4]putriutamirukmana@gmail.com
[5]nelsaputri@student.telkomuniversity.ac.id
[6]muharmanlubis@student.telkomuniversity.ac.id

***Abstract*— This research compares Naive Bayes and K-NN algorithms for predicting customer churn using a Kaggle dataset. The data preprocessing includes converting categorical variables and applying the SMOTE method for balanced data testing. Naive Bayes shows improved results on balanced data with SMOTE, while K-NN experiences a notable decrease in performance. Although K-NN maintains high accuracy (around 0.56), there are significant reductions in Precision, Recall, and F1-Score. Conversely, Naive Bayes on balanced data exhibits a decrease in F1-Score for the minority class ('exited') but maintains favorable performance. In conclusion, Naive Bayes is more robust to class imbalance than K-NN, especially with balanced data. The model choice depends on specific goals in addressing class imbalance. Further research should optimize K-NN parameters for improved performance on imbalanced data, focusing on data scale and distribution variations.**

Keywords*— *Customer churn, naïve bayes, k-nearest neigbors, machine learning, prediction, bank

I. INTRODUCTION

The banking industry has become highly competitive in recent years, making customer retention challenging [1]. Customers who persist in using the bank's services affect the long-term growth and profitability of the bank. If a bank experiences customer churn, it will cause problems such as reduced profits, and require high costs to find new customers. It is crucial for banks to know what causes customer churn in order to create effective strategies that focus on customer retention, loyalty, and sustainable growth [2].

Customer churn occurs when a customer leaves a bank and switches to a competitor or stops using banking services [3]. It is also known as customer defection or customer turnover. It is a complex phenomenon that is influenced by many factors, such as customer behavior, service quality, competitive offerings, economic circumstances, and technological advancements. Banks can gain valuable insights from knowing the causes of customer churn, which enables banks to address possible problems and improve customer satisfaction.

The emergence of machine learning technology has presented great potential in customer churn prediction in the banking sector. By using the best algorithms. Machine learning allows banks to analyze customer data in depth, for example, by analyzing variables such as credit score, account balance, and customer engagement with products and services. Machine learning algorithms can build prediction models that lead to churn, so banks can take proactive measures to retain customers [4]. This research aims to predict customer churn by utilizing machine learning. The data set used is obtained from the Kaggle data set. This data set contains details of bank customers and variables that reflect the fact whether the customer will leave the bank by closing his account or remain a customer.

The structure of this paper will be explained as follows: The first section is an introduction that describes how customer churn prediction is important in the banking industry by utilizing technology such as machine learning. The second section will explain the method used in this research. The third section explains the results and discussions that have been analyzed on the data using the naïve bayes and KNN (K-Nearest Neighbors) algorithm. The fourth section is the conclusion of this paper.

In conclusion, this study aims to provide a comprehensive overview of how to predict customer churn in the banking industry by utilizing machine learning, so as to provide valuable insights, and practical suggestions to help banks reduce the number of lost customers by increasing customer satisfaction and improving overall business performance. In an era of fierce competition, it will be crucial for banks to thrive in the market and build good customer relationships by understanding and managing customer churn effectively.

II. METHODOLOGY

This research will use the KDD (Knowledge Discovery in Database) method which has 5 stages in the process, namely: selection, pre-processing, transformation, data mining, and evaluation [5]. The following will explain some of the stages of the KDD method.

A. *Selection*

The first stage is to select relevant data because not all data will be needed in the data mining process. Data will be selected and analyzed; the data used in this research is data derived from Kaggle regarding bank customer data consisting of Row Number, Customer ID, Surname, Credit Score, Geography, Gender, Age, Tenure, Balance, number of Products, Has Cr Card, is Active Member, Estimated Salary, and the Exited label which states the customer will close the bank or not. By selecting the data to be processed, it will be better to remove irrelevant data such as Row Number, customer ID, and Surname.

B. *Pre-processing*

Most data in fact still contains noise and many missing values. Inappropriate data will impact the quality of the classifier. The pre-processing step includes data preparation and cleaning activities to accurately correct the data for further processing [6]. After checking the data using Python, the data is clean and there is no duplicate data or null data.

C. *Transformation*

At this stage, the data will be converted into a form that can be processed for the data mining process. For example, data that still uses categorical data such as Geography and Gender data is converted to numeric using Label Encoder from the Python library as in the following table.

TABLE I. BEFORE LABEL ENCODER

index	*CreditScore*	*Geography*	*Gender*
5	***850***	***Spain***	***Female***
6	645	Spain	Male
7	822	France	Male
8	376	Germany	Female
9	501	France	Male

In this example, we will convert the Gender and Geography columns to numeric form using Label Encoder. In the Gender column, Female will be replaced with the value 0, while Male will be replaced with the value 1. In the Geography column, the value France will be represented by the number 0, Germany by the number 1, and Spain by the number 2.

TABLE II. AFTER LABEL ENCODER

index	*Credit Score*	*Geography*	*Gender*
5	***850***	***2***	***0***
6	645	2	1
7	822	0	1
8	376	1	0
9	501	0	1

After the Label Encoder process, the results can be seen in Table 2, which is a comparison of the results before and after the transformation. With this transformation, categorical data in string form has been converted into a numerical form that can be used in data mining processes. Furthermore, after the data transformation is complete, the data is ready for further processing using various data mining techniques, such as modeling customer churn predictions.

D. *Data Mining*

The data mining stage is modeling by predicting customer churn on data that has been selected, preprocessed and transformed. However, according to the results of the data analysis, the data has an imbalance problem for the classes, so it requires balancing the data for each class using the Synthetic Minority Over-sampling Technique (SMOTE) library from learn. So oversampling will be conducted so that the minority class will be as large as the majority class. In Fig. 1 and Fig. 2 illustrate the imbalance data and balanced data using SMOTE.

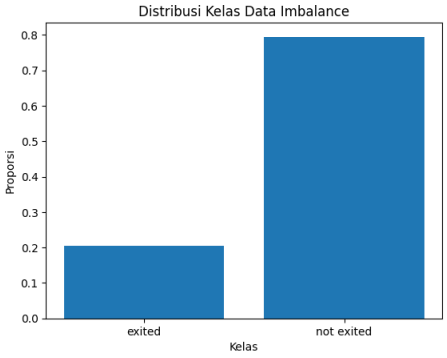

Fig. 1. Imbalance Data

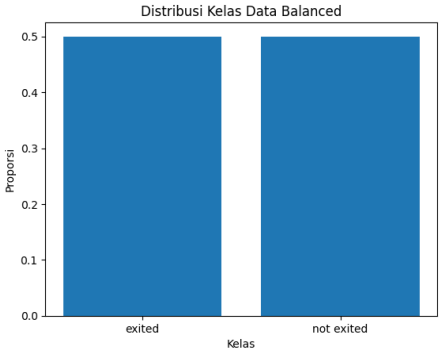

Fig. 2. Balanced Data

After dealing with the unbalanced data, the next step is to divide the data into training and testing data with a ratio of 80:20 using the train test split from scikit-learn. Then the next step is to perform modelling using the Naïve Bayes algorithm provided by the scikit-learn library in Python.

Naive Bayes is based on Bayes' theorem and is known as a probabilistic classifier, using strong independent

assumptions, and independent feature models. This research will use Gaussain Naive Bayes, where continuous values for each class are distributed according to a normal distribution [7].

E. Evaluation

At this stage, the confusion matrix is used for testing and evaluation of the model that has been made at the data mining stage. The evaluation here will look at the accuracy, recall, precision, and f1-score values of the model.

Confusion matrix is a matrix that contains information about the actual class and predicted class obtained from the algorithm [8]. Confusion matrix can be used to evaluate the quality of modelling, for example for problems in two classes using the results of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [9]. Here are some equations to calculate accuracy, recall, precision and f1-score.

$$accuracy = \frac{TP + TN}{P + N} \quad (1)$$

$$recall = \frac{TP}{P} \quad (2)$$

$$precision = \frac{TP}{TP + FP} \quad (3)$$

$$f1 - score = \frac{2 . precision . recall}{precision + recall} \quad (4)$$

Accuracy measures the extent to which the model can correctly classify the data as a whole. Precision measures the extent to which the model provides correct predictions for positive classes, while recall measures the extent to which the model can identify all true positive data. F1 score combines precision and recall to provide a balanced measure. Understanding this matrix is important to evaluate the performance of a classification model and gain a holistic understanding of the model's ability to correctly classify data.

III. RESULT AND DISCUSSION

All implementations carried out in this research use Google Collab tools and the sci-kit-learn library. This research will explain some of the results of the implementation carried out by knowing some of the variables that affect the prediction results such as Credit Score, Age, and Balance as follows.

A. Data Analysis

This research will analyze predictions based on the variables that have been selected and will be analyzed based on the Credit Score, Age, and Balance variables. There are several figures to see the overall data about the three variables. In Fig. 1. Illustrates that most of the customers have a Credit Score with the highest range of 600 to 700 with an average of 650.5288. So it can be characterized that customers with a range of 660 and above are quite good in credit scores at the bank, and what should be watched out for is the credit score range of 660 and below. In Fig.4. depicts the ages that use the bank's services the most, with an average age of 38 being bank customers. While Fig.5. illustrates the balance that customers save in the bank, most customers still have a balance of 0, and have an average balance of 76485.89.

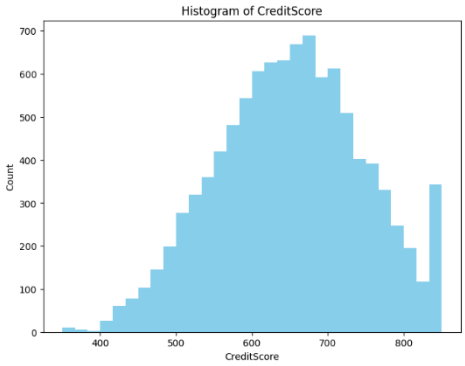

Fig. 3. Histogram CreditScore

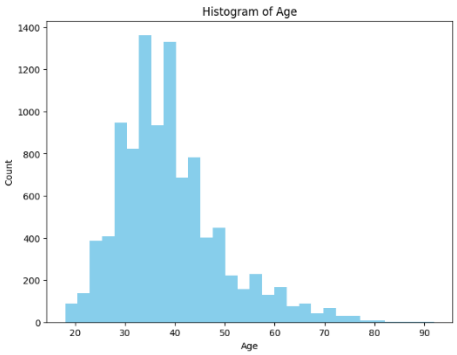

Fig. 4 Histogram Age

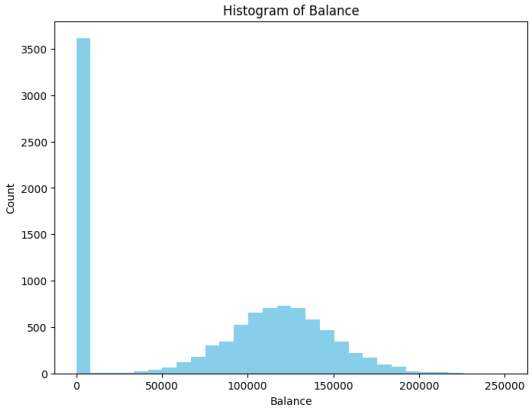

Fig. 5. Histogram Balance

B. Units Customer Churn Prediction (Imbalance and Balanced Data)

After analyzing the data and getting an overview of several variables to be predicted, the next step is to predict customer churn using the Naive Bayes and KNN (K-Nearest Neighbors)

algorithm. The following are the results of the classification report on unbalanced and balanced data using the Naive Bayes algorithm.

TABLE III. CLASSIFICATION REPORT IMBALANCE DATA NAÏVE BAYES

	Precision	***Recall***	***F1-Score***
not exited	***0,82***	***0,97***	***0,89***
exited	0,47	0,10	0,17
Accuracy			0,80

TABLE IV. CLASSIFICATION REPORT BALANCED DATA (NAÏVE BAYES)

	Precision	***Recall***	***F1-Score***
not exited	***0,93***	***0,71***	***0,81***
exited	0,40	0,77	0,52
Accuracy			0,73

Based on the mentioned results, it can be observed that when analyzing unbalanced data, the model has high accuracy. However, accuracy alone does not give a complete picture of the model's performance. Therefore, it is important to look at other metrics such as precision, recall, and f1-score.

On unbalanced data, the classification report results show that the model has high accuracy, but the precision, recall, and f1-score values for the predicted exited classes are very low. This shows that while the model can classify the majority of the data correctly, its ability to identify customers who will actually exit is low. In this context, precision measures the extent to which customers predicted to be exited are actually exited, while recall measures the extent to which the model can identify all customers who are actually exited.

The following are the results of the classification report on unbalanced and balanced data using the K-NN algorithm.

TABLE V. CLASSIFICATION REPORT IMBALANCE DATA (K-NN)

	Precision	***Recall***	***F1-Score***
not exited	***0,81***	***0,93***	***0,86***
exited	0,24	0,09	0,14
Accuracy			0,76

TABLE VI. CLASSIFICATION REPORT BALANCED DATA (K-NN)

	Precision	***Recall***	***F1-Score***
not exited	***0,80***	***0,60***	***0,69***
exited	0,20	0,40	0,27
Accuracy			0,56

However, when using data processing techniques such as Synthetic Minority Over-sampling Technique (SMOTE) to overcome data imbalance, the classification report results show an increase in precision, recall, and f1-score. This means that the model has a better ability to correctly identify customers who will be exited. SMOTE helps in creating synthetic samples for the minority class, thus improving the data balance and allowing the model to learn better from both classes.

C. Confusion Matrix (Imbalance dan Balanced Data)

After analyzing the classification report results based on imbalance and balanced data. Then the following are the results of the confusion matrix given on imbalance data, as follows:

- The number of True Positives (TP) is 41, which is the number of samples that are actually predicted as churn by the model.
- The number of False Positives (FP) is 47, which is the number of samples that are incorrectly predicted as churn by the model, when they should not be churn.
- The number of False Negatives (FN) is 352, which is the number of samples that are incorrectly predicted as not churning by the model, when they should be churning.
- The number of True Negatives (TN) is 1560, which is the number of samples that are actually predicted as not churning by the model.

In the context of predicting customer churn, these results show that the model has a significant error rate in predicting churn. There is a fairly high number of False Positives (FP), which predicts customers will churn when they are not. In addition, the number of False Negatives (FN) is also quite large, that is, there are many churn cases that are missed and predicted as not churn. By analyzing the precision, recall, and f1-score values, the precision value of about 46.57% indicates that of all those predicted as churn, only a small percentage are actually churn. Recall of about 10.44% indicates that the model has difficulty in detecting actual churn cases. And the f1-score value is about 16.72% which describes the balance between precision and recall, and the value shows that the model has difficulty in predicting churn well.

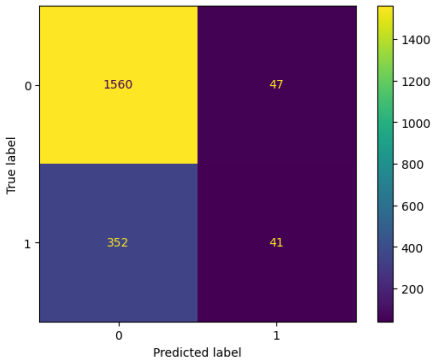

Fig. 6. Confusion Matrix Imbalance Data with Naive Bayes

Furthermore, the results of the confusion matrix given on the balanced data are as follows:

- The value of True Positives (TP) is 302, which is the number of samples that are actually predicted as churn by the model.
- The value of False Positives (FP) is 458, which is the number of samples that are incorrectly predicted as churn by the model, when they should not be churn.
- The value of False Negatives (FN) is 91, which is the number of samples that the model incorrectly predicted as not churning, when they should be churning.
- The value of True Negatives (TN) is 1149, which is the number of samples that are actually predicted as not churning by the model.

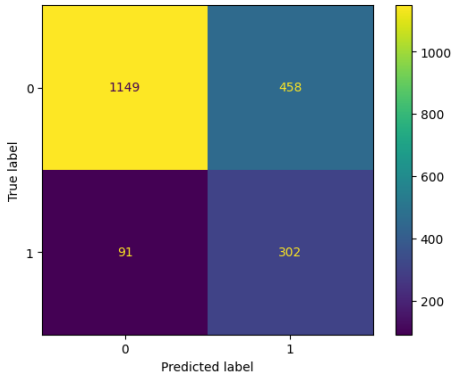

Fig. 7. Confusion Matrix Balanced Data with Naive Bayes

From these results, further analysis can be conducted. Although the accuracy of the model decreased when compared to the accuracy of the imbalanced data model, decreasing by about 7% with an accuracy of 73%, however, there is a significant difference between recall and f1-score. Recall, which measures the extent to which the model can correctly identify all positive (churn) cases, has a value of about 76.89%. This means that the model can detect most of the churn cases, but there are still a small number of churn cases that are missed and predicted as not churn.

This shows that the model has a tendency to be better at identifying customers who do not churn, but still needs to improve its ability to recognize customers who will actually churn. In the context of customer churn prediction, it is very important to reduce false positives and false negatives in order to take appropriate actions to retain customers who intend to churn and reduce unnecessary churn.

Other analysis results using K-NN based on imbalanced data obtained results in the confusion matrix, with the following values:

- The value for True Positive (TP) is 1491. These are situations where the model predicted a positive outcome and it proved correct.
- The value for False Positive (FP) is 116. This reflects instances where the model predicted a false positive.
- The value for False Negative (FN) is 356. These are false negatives, which occur when the model fails to recognize a good outcome.
- The value for True Negative (TN) is 37. These are situations where the model predicted a negative outcome and the reality was indeed negative.

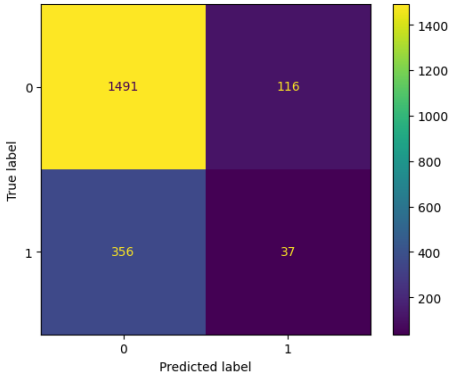

Fig. 8. Confusion Matrix Imbalance Data with K-NN

For the results of balanced data using K-NN, the results for the confusion matrix value are as follows:

- True Positive (TP), refers to 969 cases when the KNN algorithm correctly recognized positive cases.
- False Positive (FP), there are 638 observations that should have been classified as negative but were incorrectly classified as positive.
- False Negative (FN): 235 observations that should have been classified as positive were incorrectly classified as negative.
- True Negative (TN): 158 observations were correctly classified as negative.

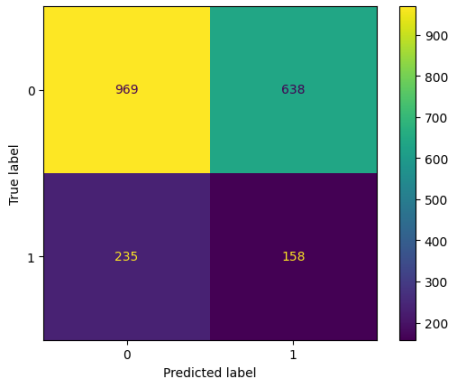

Fig. 9. Confusion Matrix Balanced Data with K-NN

In conclusion, although the accuracy of the model appears high, it is important to look at other evaluation metrics such as precision, recall and f1-score. Improvements in recognizing churn cases more accurately will help banks take appropriate actions to retain customers and optimize retention strategies.

D. Analysis Result

Based on the analysis, the Naive Bayes algorithm has the highest results in accuracy and other matrix evaluation values.

It can be concluded that in the group of customers who are predicted to leave (Exited), the average age is 41 years, with Credit Score most often having a value of 850. The highest balance predicted to exit is 216,109.88, while the lowest balance is 0.0. In terms of gender, female customers are more often predicted to be Exited. Meanwhile, in the Not Exited group, the average age is 33 years old, with a Credit Score that most often has a value of 850. The highest balance predicted to not exit is 209,490.21, while the lowest balance is 0.0. In terms of gender, male customers are more likely to be predicted as Not Exited.

There are differences in age, Credit Score, balance, and gender characteristics between the groups of customers predicted to exit and not exit. This information can be used as a guide to identify factors that influence customers' decisions to leave or not leave the bank, so that banks can take appropriate actions to maintain customer loyalty and reduce churn.

IV. CONCLUSION

This research uses the Naive Bayes and K-NN (K-Nearest Neigbors) classification method on the Kaggle dataset to perform customer drop-off analysis and prediction. In the data pre-processing stage, string variables such as Geography and Gender are converted into numbers using Label Encoder. Next, the SMOTE method was used to test the unbalanced and balanced data.

Based on the results of research conducted on these two algorithms, Naïve Bayes has improved results on balanced data with SMOTE, but the K-NN model has experienced a significant decrease in performance. Although it still has relatively high accuracy (around 0.56), there is a significant decrease in Precision, Recall, and F1-Score for both classes. In contrast, the Naive Bayes model on balanced data shows a decrease in F1-Score for the minority class ('exited'), but still maintains relatively good performance with Precision around 0.40, Recall around 0.77, and F1-Score around 0.52 . In conclusion, in this case, Naive Bayes seems to be more robust to class imbalance than K-NN, especially when the data is transformed to be balanced. The choice of model depends on the specific goals and priorities in dealing with class imbalance in a classification task.

In the context of a comparison between K-NN and Naive Bayes, it can be concluded that K-NN has high sensitivity to different data scales and distributions, while Naive Bayes is more tolerant of these variations. Therefore, in future research, it is recommended to optimize K-NN parameters, especially the number of neighbors and distance metrics, to improve performance on imbalanced data. Focusing on addressing the issues of different data scales and distributions can help improve the robustness of K-NN models to variations in the dataset.

REFERENCES

[1] H. Sun *et al.*, "CSR, Co-Creation and Green Consumer Loyalty: Are Green Banking Initiatives Important? A Moderated Mediation Approach from an Emerging Economy," *Sustainability*, vol. 12, no. 24, p. 10688, Dec. 2020, doi: 10.3390/su122410688.

[2] S. Ghamry and H. M. Shamma, "Factors influencing customer switching behavior in Islamic banks: evidence from Kuwait," *Journal of Islamic Marketing*, vol. 13, no. 3, pp. 688–716, Feb. 2022, doi: 10.1108/JIMA-01-2020-0021.

[3] P. Lalwani, M. K. Mishra, J. S. Chadha, and P. Sethi, "Customer churn prediction system: a machine learning approach," *Computing*, vol. 104, no. 2, pp. 271–294, Feb. 2022, doi: 10.1007/s00607-021-00908-y.

[4] E. Domingos, B. Ojeme, and O. Daramola, "Experimental Analysis of Hyperparameters for Deep Learning-Based Churn Prediction in the Banking Sector," *Computation*, vol. 9, no. 3, p. 34, Mar. 2021, doi: 10.3390/computation9030034.

[5] S. Alam, M. G. Resmi, and N. Masripah, "Classification of Covid-19 vaccine data screening with Naive Bayes algorithm using Knowledge Discovery in database method," *Journal of Computer Networks, Architecture and High Performance Computing*, vol. 4, no. 2, pp. 177–185, Jul. 2022, doi: 10.47709/cnahpc.v4i2.1584.

[6] C. Wagner, P. Saalmann, and B. Hellingrath, "Machine Condition Monitoring and Fault Diagnostics with Imbalanced Data Sets based on the KDD Process," *IFAC-PapersOnLine*, vol. 49, no. 30, pp. 296–301, 2016, doi: 10.1016/j.ifacol.2016.11.151.

[7] S. B. S. Mugdha *et al.*, "A Gaussian Naive Bayesian Classifier for Fake News Detection in Bengali," 2021, pp. 283–291. doi: 10.1007/978-981-33-4367-2_28.

[8] H. Ahmed and A. K. Nandi, "Classification Algorithm Validation," in *Condition Monitoring with Vibration Signals*, Wiley, 2019, pp. 307–319. doi: 10.1002/9781119544678.ch15.

[9] J. Han, "Data Mining Third Edition," 2000.