JURNAL INFORMATIKA
PROXIES

Vol.9, No.1, July, 2025

The Effect Of Chi-Square Feature Selection On The Naive Bayes Algorithm In Analyzing The Sentiment Of Gojek Application Reviews On Google Play Store

Rafael Handika Dwinanta[1], Shinta Estri Wahyuningrum[2]

1 Teknik Informatika Fakultas Ilmu Komputer, Universitas Katolik Soegijapranata, 21k10028@student.unika.ac.id

2 Teknik Informatika Fakultas Ilmu Komputer, Universitas Katolik Soegijapranata, shinta@unika.ac.id

Corresponding Author Email: shinta@unika.ac.id

https://doi.org/10.24167/proxies.v9i1.13124

Received: 2025-01-30
Revised: 2025-07-05
Accepted: 2025-10-22
Available online: 2025-10-23

Keywords:
sentiment analysis, Naive Bayes, Chi-Square, feature selection, Gojek reviews

ABSTRACT

This study analyzes customer sentiment in reviewing the Gojek application to find out whether Chi-Square feature selection can improve the performance of the sentiment analysis model. This study uses 12,000 Gojek review data, starting with labeling positive, negative, or neutral based on user ratings of the reviews. Naive Bayes with and without Chi-Square feature selection is used in testing related to accuracy, precision, recall, and F1 score. The best performance is obtained by using alpha 0.5 combined with the best 2000 Chi-Square features, which produces 86.96% accuracy, 87.84% precision, 86.96% recall, and 85.29% F1 score on imbalanced data. SMOTE is also used to handle the low number of neutral reviews, but it produces lower accuracy. In conclusion, Chi-Square feature selection in the Naive Bayes algorithm can improve model accuracy on imbalanced and balanced datasets

1. INTRODUCTION

We live in an era where technology advancement and information have overlaid massive effects on our daily lives. Today, different activities have been simplified by online-based applications. The majority of the population relies on technology to help them with the various daily activities, such as booking transportation online. One example of such an application is Gojek. Currently, Gojek is a very popular application with many reviews. The quality of the Gojek application can be evaluated based on user ratings and reviews in Google Play Store. Gojek faces difficulty in manually analyzing the sentiment of such many reviews.

Some of the commonly used algorithms in analyzing the feedback by users include Naive Bayes, SVMs which in turn find application in transportation-based apps, e-commerce portals, and educational domains [1-7]. Various works emphasize that stemming, stop-word removal, and tokenization are important preprocessing steps that are vital for higher precision in analysis [1, 8-10]. Feature selection techniques such as Chi-Square also emerge, showing how these methods enhance the performance of a model by selecting only the most relevant features [11, 12]. Besides, the issue of class imbalance is being resolved with the help of oversampling techniques, which includes SMOTE [13, 14] for improving classification performance and ensuring its reliability.

This research was conducted to compare Naive Bayes algorithm with and without Chi-Square feature selection in order to find out Chi-Square feature selection effect in analyzing sentiment using Gojek application reviews based on highest accuracy, precision, recall, and f1-score values.

2. RESEARCH METHODOLOGY

The methodology includes steps from data collection to testing the model. Detailed descriptions of each step are given to make sure the research process is clear and can be repeated accurately. Research methodology of this research is presented in Figure 1.

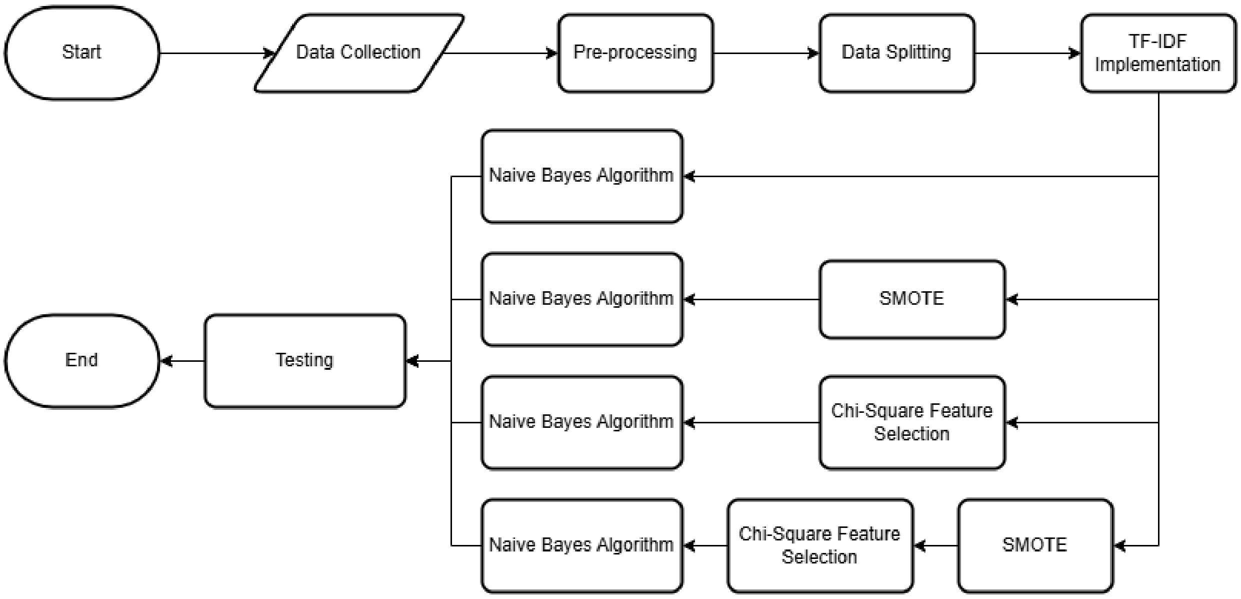

Figure 1. Flowchart Research Methodology

The research begins with Data Collection, Pre-processing, Data Splitting, TF-IDF Implementation, Algorithm Implementation (using Naïve Bayes algorithm with and without Chi-Square feature selection on imbalanced and balanced data), and then testing to find out effect of Chi-Square feature selection for analyzing sentiment.

2.1 Dataset Collection

In this research, dataset collection is conducted by taking datasets from the Kaggle website regarding Gojek application reviews on the Google Play Store from 2021 to 2024. The dataset has five columns, which include usernames, content, score, at, and appVersion. For this research, only content and score parameters are used to analyze sentiment. The dataset is already in csv form, so it can be directly used in Google Colab. Total number of datasets is 225,043 data, but only the first 12,000 data used. Dataset can accessed from link: https://www.kaggle.com/datasets/ucupsedaya/gojek-app-reviews-bahasa-indonesia.

2.2 Pre-processing

2.2.1 Data Labeling

Data labeling is the process of assigning sentiment labels to user reviews based on given scores. Reviews gathered in this research from the Gojek application in the Google Play Store were labeled into three, namely: positive, neutral, and negative sentiments, each depending on the score given to the review. Data labeling is needed in supervised learning, where the machine learning model will be trained with labeled data to predict the sentiment of new unseen reviews.

2.2.2 Tokenizing

Tokenizing is breaking down text process to smaller units, so that they can processed and analyzed by a computer. In this research, tokenizing involves splitting review comments word by word using the nltk library. Every pre-processing step requires tokenizing to ensure the data is in a form that can be used for further analysis.

2.2.3 Stopword Removal

Stopword removal filtration of common words that always appear repeatedly in the text with very little meaningful information. Examples of such words in the Indonesian language include "yang", "dan", "di", and "dari". The purpose of stop-word removal is to get rid of these low-information words so that attention can be maintained on more significant terms that contribute to sentiment analysis.

2.2.4 Stemming

Stemming is one of data pre-processing steps which reduces words to their root form. This step is crucial because it helps us to standardize words by converting different word form into one form, which simplifies analysis. For example, words "berlari", “berlarian" would be reduced to their root form "lari".

2.3 Data Splitting

Data splitting is an integral part of data preparation prior to its use in machine learning. Basically, the credence is chopped into different subsets used in serving the modeling workflow purposes. Most of the time, data will either be split into a train and test dataset, with the former holding the model during training and the latter testing or evaluating it.

2.4 TF-IDF Implementation

Term Frequency-Inverse Document Frequency (TF-IDF) is a metric used to measure the importance of a word in a document against a set of documents. TF-IDF value increases proportional to the number of times a word appears in document but is balanced by word frequency in corpus, helping adjust for the fact that some words occur more often in common.

Term Frequency (TF) can be calculated by:

$$\text{TF}(t, d) = \frac{\textit{Frequency of term t in document d}}{\textit{Total number of terms in document d}} \tag{1}$$

Inverse Document Frequency (IDF) can be calculated by:

$$\text{IDF}(t, D) = \log\left(\frac{\textit{Total number of documents}}{\textit{Number of documents containing term t}}\right) \tag{2}$$

TF-IDF can be calculated by:

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D) \tag{3}$$

2.5 Synthetic Minority Over-sampling Technique (SMOTE)

SMOTE is a technique to handle class imbalance by creating new synthetic samples for the minority class, thus making it closer in number to the majority class. SMOTE selects samples in the minority class that are in proximity to each other and creates synthetic samples between those. This technique follows the conversion of the text data into numerical features using TF-IDF vectorization.

2.6 Chi-Square Feature Selection

Chi-Square feature selection is a statistical method for the selection of relevant variables in a given classification problem. In text classification, it is about placing a class most surely with the indicative words of every class, improving model performance through reducing dimensionality in the feature space and focusing on the most informative features.

Expected frequency for each class can be calculated by:

$$Eij = \frac{(total\ documents\ in\ class\ j) \times (total\ word\ occurrences\ of\ word\ i)}{total\ documents} \tag{4}$$

Chi-Square can be calculated by:

$$x^2 = \Sigma \frac{(Oij - Eij)^2}{Eij} \tag{5}$$

2.7 Naïve Bayes Algorithm

These filtered features from Chi-Square were used directly for the training of Naive Bayes. By training the model with only top-ranked features, Chi-Square keeps the algorithm away from getting biased towards any irrelevant or noisy features, thus improving its generalization capability. After Chi-square feature selection, the TF-IDF matrix of both the training and testing dataset is reduced to only top-ranked features. Chi-Square feature selection in Naive Bayes lets the model give focus to the most influential features, hence preventing overfitting and improving the performance of the model.

Calculate the likelihood for each word given each class using Laplace smoothing to avoid zero probabilities.

$$\text{P}(\boldsymbol{word}|\boldsymbol{class}) = \frac{(\boldsymbol{count}(\boldsymbol{word\ in\ class}) + \mathbf{1})}{(\boldsymbol{total\ words\ in\ class} + \boldsymbol{vocab\ size})} \tag{6}$$

2.8 Testing

In this step, testing will calculate the accuracy, precision, recall, and f1-score values according to the confusion matrix which is obtained from the result using Naive Bayes algorithm with and without Chi-Square feature selection. Testing using confusion matrix method as below:

a. Accuracy

Accuracy can be calculated by:

$$\frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (7)$$

b. Precision
Precision can be calculated by:

$$\frac{TP}{TP + FP} \times 100 \quad (8)$$

c. Recall
Recall can be calculated by:

$$\frac{TP}{TP + FN} \times 100 \quad (9)$$

d. F1-Score
F1-Score can be calculated by:

$$\frac{2 \times (recall \times precision)}{(recall + precision)} \quad (10)$$

3. RESULT AND DISCUSSION

3.1 Result

The performance evaluation is organized into three main sections. The first section involves finding the best alpha value to get the optimal performance of the model. The second section compares Naive Bayes performance on imbalanced and balanced data using SMOTE to handle class imbalances. The third section focuses on using Chi-Square feature selection to enhance model results by selecting the relevant features. The third section is divided into two parts: finding the best Chi-Square configuration and comparing the performance of models, which were trained with best Chi-Square features on both imbalanced and balanced datasets.

3.1.1 Finding the Optimal Alpha Value

Alpha adjusts how the model deals with words that appear rarely in the data. To improve the Naive Bayes model, different values of the alpha hyperparameter were tested, specifically values between 0.1 and 1.0. This setting helps prevent the model from giving too much importance to unusual words, which can make the model perform better. The optimal alpha was identified based on performance across accuracy, precision, recall, and F1-score.

Table 1. Comparison of Naive Bayes Results Based on Different Alpha Values

No	Alpha	Accuracy	Precision	Recall	F1-Score
1	0.1	0.8504	0.8315	0.8504	0.8398
2	0.2	0.8587	0.8328	0.8587	0.8434
3	0.3	0.8612	0.8292	0.8612	0.8443
4	0.4	0.8620	0.8290	0.8620	0.8445
5	0.5	0.8629	0.8296	0.8629	0.8453
6	0.6	0.8620	0.8287	0.8620	0.8445
7	0.7	0.8612	0.8276	0.8612	0.8436
8	0.8	0.8616	0.8277	0.8616	0.8439
9	0.9	0.8608	0.8265	0.8608	0.8429
10	1.0	0.8608	0.8264	0.8608	0.8429

Table 1 provides the performance metrics of the Naive Bayes algorithm at various alpha values ranging from 0.1 to 1.0. The results indicate that alpha = 0.5 yields the highest accuracy at 0.8629, along with a precision of 0.8296, recall of 0.8629, and F1-score of 0.8453. As alpha increases from 0.1 to 0.5, there is a gradual improvement in accuracy, suggesting that moderate smoothing (alpha = 0.5) helps balance the handling of frequent and rare terms effectively. However, when alpha values exceed 0.5, the accuracy and other metrics begin to slightly decrease, which may be due to over-smoothing, reducing the distinctiveness of some features and slightly lowering the model's effectiveness.

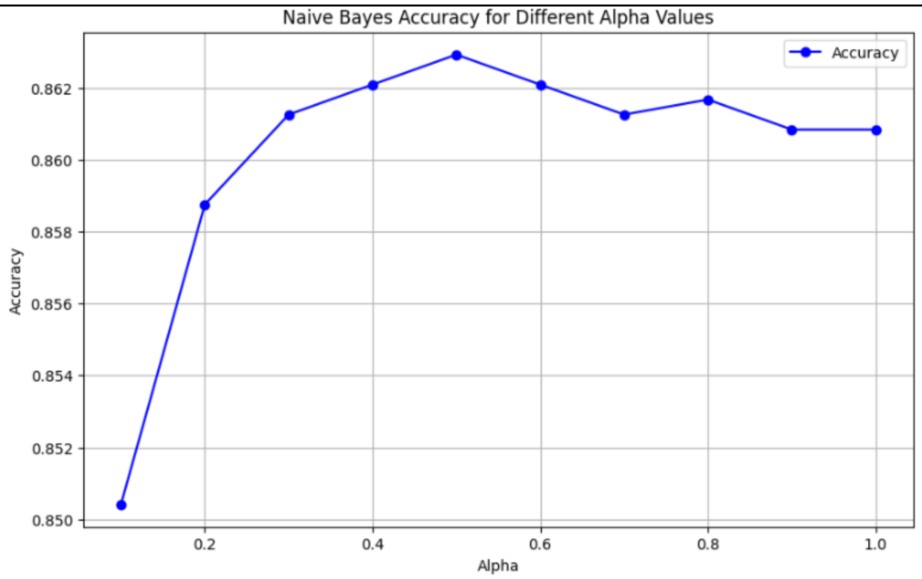

Figure 2. Naive Bayes Accuracy for Different Alpha Values

Figure 2 visually demonstrates the effect of varying alpha values on accuracy. The graph shows a clear peak at alpha = 0.5, where accuracy is highest. This figure supports the selection of alpha = 0.5 as the most optimal alpha parameter for the 12000 dataset.

3.1.2 Naïve Bayes Performance on Imbalanced and Balanced Data

This section examines the performance of the Naive Bayes algorithm on imbalanced and balanced data, both using an alpha value of 0.5, identified as optimal in the previous section. SMOTE was applied to balance the dataset, equalizing the distribution of positive, negative, and neutral classes.

Table 2. Comparison of Naive Bayes Results on Imbalanced and Balanced Data

No	Method	Accuracy	Precision	Recall	F1-Score
1	Imbalanced (without SMOTE)	0.8629	0.8296	0.8629	0.8453
2	Balanced (with SMOTE)	0.8151	0.8192	0.8151	0.8156

Table 2 shows that without SMOTE, the model achieves a higher accuracy of 0.8629, precision of 0.8296, recall of 0.8629, and F1-score of 0.8453. In contrast, applying SMOTE reduces accuracy to 0.8151, with precision, recall, and F1-score decreasing slightly as well. Although SMOTE rebalanced the classes and improved the model's ability to recognize minority classes, this led to an overall decline in performance metrics. The potential reason for this decreasing could be overfitting, as the synthetic samples created by SMOTE may introduce noise or irrelevant features that affect model generalization.

3.1.3 Naïve Bayes with Chi-Square

In this section, the impact of Chi-Square feature selection on the Naive Bayes classifier is explored. Chi-Square is a statistical method used to select features that have the highest relevance to the target variable, in this case, sentiment categories (positive, negative, and neutral). By reducing the dimensionality of the dataset, Chi-Square aims to improve the model’s efficiency and potentially its accuracy by focusing on the most significant features.

a. Finding the Optimal Chi-Square Feature on Imbalanced and Balanced Data

This section aims to identify the optimal number of Chi-Square features for sentiment classification using the Naive Bayes algorithm on imbalanced and balanced dataset. The imbalanced dataset uses the original data distribution, while the balanced dataset is created by applying SMOTE to equalize the class distribution among positive, negative, and neutral classes. Each configuration is evaluated based on accuracy, precision, recall, and F1-score.

Table 3. Chi-Square Feature Results on Imbalanced Data

No	Chi-Square Features	Accuracy	Precision	Recall	F1-Score
1	100	0.8370	0.8000	0.8370	0.8175
2	500	0.8616	0.8704	0.8616	0.8451
3	1000	0.8654	0.8744	0.8654	0.8488
4	2000	0.8695	0.8783	0.8695	0.8529
5	3000	0.8650	0.8731	0.8650	0.8482
6	4000	0.8633	0.8713	0.8633	0.8459
7	5000	0.8625	0.8283	0.8625	0.8446

In Table 3, the best performance on the imbalanced data is achieved with 2000 Chi-Square features, resulting in an accuracy of 0.8695, precision of 0.8783, recall of 0.8695, and F1-score of 0.8529. As seen in the table, the accuracy improves as Chi-Square features increase from 100 to 2000, but declines with additional features. This indicates that adding too many features beyond 2000 introduces irrelevant or redundant information, which slightly reduces the model's performance.

Table 4. Chi-Square Feature Results on Balanced Data

No	Chi-Square Features	Accuracy	Precision	Recall	F1-Score
1	100	0.6316	0.6210	0.6316	0.6191
2	500	0.7466	0.7483	0.7466	0.7451
3	1000	0.7942	0.8004	0.7942	0.7937
4	2000	0.8242	0.8288	0.8242	0.8242
5	3000	0.8327	0.8365	0.8327	0.8327
6	4000	0.8382	0.8411	0.8382	0.8382
7	5000	0.8371	0.8395	0.8371	0.8371

On the balanced dataset, shown in Table 4, the optimal performance is reached with 4000 Chi-Square features, achieving an accuracy of 0.8382, precision of 0.8411, recall of 0.8382, and F1-score of 0.8382. Accuracy and other metrics increase with the number of Chi-Square features up to an optimal point (4000 features), after which the performance decreases slightly.

b. Comparison of Optimal Chi-Square Feature on Imbalanced and Balanced Data

In this section, Naive Bayes was tested using 2000 Chi-Square features on the imbalanced data (without SMOTE) and 4000 Chi-Square features on the balanced data (with SMOTE).

Table 5. Comparison of Optimal Chi-Square Feature on Imbalanced and Balanced Data

No	Method	Chi-Square Features	Accuracy	Precision	Recall	F1-Score
1	Imbalanced (without SMOTE)	2000	0.8695	0.8783	0.8695	0.8529
2	Balanced (with SMOTE)	4000	0.8382	0.8411	0.8382	0.8382

Table 5 summarizes that the Naive Bayes model performs better on the imbalanced data with 2000 Chi-Square features, achieving an accuracy of 0.8695. In comparison, balancing the data using SMOTE with 4000 Chi-Square features lowers the accuracy to 0.8382, with precision, recall, and F1-score all slightly lower as well. This result suggests that the balanced dataset, even with optimal Chi-Square features, does not surpass the performance on the imbalanced data

3.2 Discussion

The first experiment to determine the optimal alpha value found that alpha 0.5 gave the highest score, achieving an accuracy of 0.8629. This moderate alpha value balances the influence of common and rare words.

In the comparison of imbalanced and balanced data, the use of SMOTE to balance the classes did not improve accuracy. The accuracy on balanced data reached 0.8151, slightly lower than imbalanced data with a result of 0.8629. This shows that although SMOTE effectively balances the class distribution, it can also cause noise in the data by creating synthetic samples that are less suitable for sentiment. Previous studies have shown that SMOTE can improve the representation of minority classes, but can be at risk of overfitting [14].

The selection of Chi-Square features gave the highest accuracy on imbalanced data with 2000 features, resulting in an accuracy of 0.8696. On balanced data with SMOTE, the optimal features were obtained at 4000 features, resulting in an accuracy of 0.8383. This result shows that Chi-Square effectively improves model performance by filtering out less relevant terms, allowing Naive Bayes to focus on more meaningful features [5].

4. CONCLUSION

The experiments in this research show that Naive Bayes can effectively classify sentiment in Gojek app reviews, especially when Chi-Square feature selection is applied. The highest performance was achieved on imbalanced data with 2000 Chi-Square features with an accuracy of 0.8696.

Chi-Square feature selection has a significant impact on model accuracy, especially with 2000 features on imbalanced data and 4000 features on balanced data with SMOTE. When the number of features exceeds these values, accuracy begins to decline. This indicates that too many features actually produce irrelevant information. Thus, Chi-Square proves useful in filtering out unimportant words, allowing the model to focus on the features that are most relevant to sentiment classification.

Applying SMOTE to balanced data reduces accuracy to 0.8151. These results indicate that although SMOTE effectively balances classes, it can also add synthetic data points that do not accurately represent the true patterns in the dataset, which causes accuracy to decrease.

Future research could explore additional feature selection techniques, such as Information Gain, to see if they further improve model performance. Additionally, testing alternative balancing approaches such as ADASYN or a combined SMOTE and under-sampling method may be able to address the overfitting that occurred in this study.

REFERENCES

[1] Iwan Sinanto Ate and Ahlijati Nuraminah, “Komparasi Algoritma Feature Selection Pada Analisis Sentimen Review Film,” JUITIK, vol. 2, no. 2, pp. 96–102, Jul. 2022, doi: 10.55606/juitik.v2i2.326.

[2] A. P. P. Wardani, A. Adiwijaya, and M. D. Purbolaksono, “Sentiment Analysis on Beauty Product Review Using Modified Balanced Random Forest Method and Chi-Square,” josh, vol. 4, no. 1, pp. 1–7, Oct. 2022, doi: 10.47065/josh.v4i1.2047.

[3] K. D. Indarwati and H. Februariyanti, “Analisis Sentimen Terhadap Kualitas Pelayanan Aplikasi Gojek Menggunakan Metode Naive Bayes Classifier,” JATISI, vol. 10, no. 1, Mar. 2023, doi: 10.35957/jatisi.v10i1.2643.

[4] W. Yulita, “Analisis Sentimen Terhadap Opini Masyarakat Tentang Vaksin Covid-19 Menggunakan Algoritma Naïve Bayes Classifier,” JDMSI, vol. 2, no. 2, p. 1, Aug. 2021, doi: 10.33365/jdmsi.v2i2.1344.

[5] M. B. Hamzah, “Classification of Movie Review Sentiment Analysis Using Chi-Square and Multinomial Naïve Bayes with Adaptive Boosting,” J. Adv. Inf. Syst. Tech, vol. 3, no. 1, pp. 67–74, Apr. 2021, doi: 10.15294/jaist.v3i1.49098.

[6] H. Setiawan, E. Utami, and S. Sudarmawan, “Analisis Sentimen Twitter Kuliah Online Pasca Covid-19 Menggunakan Algoritma Support Vector Machine dan Naive Bayes,” JKKI, vol. 5, no. 1, pp. 43–51, Jul. 2021, doi: 10.31603/komtika.v5i1.5189.

[7] M. D. Hendriyanto, A. A. Ridha, and U. Enri, “Analisis Sentimen Ulasan Aplikasi Mola Pada Google Play Store Menggunakan Algoritma Support Vector Machine,” INTECOMS, vol. 5, no. 1, pp. 1–7, Apr. 2022, doi: 10.31539/intecoms.v5i1.3708.

[8] A. P. Giovani, A. Ardiansyah, T. Haryanti, L. Kurniawati, and W. Gata, “ANALISIS SENTIMEN APLIKASI RUANG GURU DI TWITTER MENGGUNAKAN ALGORITMA KLASIFIKASI,” JTI, vol. 14, no. 2, p. 115, Jul. 2020, doi: 10.33365/jti.v14i2.679.

[9] Universitas Muria Kudus and F. Rizqi Irawan, “ANALISIS SENTIMEN TERHADAP PENGGUNA GOJEK MENGGUNAKAN METODE K-NEARSET NEIGHBORS,” JIKO, vol. 5, no. 1, pp. 62–68, Apr. 2022, doi: 10.33387/jiko.v5i1.4267.

[10] A. R. Isnain, H. Sulistiani, B. M. Hurohman, A. Nurkholis, and S. Styawati, “Analisis Perbandingan Algoritma LSTM dan Naive Bayes untuk Analisis Sentimen,” JEPIN, vol. 8, no. 2, p. 299, Aug. 2022, doi: 10.26418/jp.v8i2.54704.

[11] T. Ernayanti, M. Mustafid, A. Rusgiyono, and A. R. Hakim, “PENGGUNAAN SELEKSI FITUR CHI-SQUARE DAN ALGORITMA MULTINOMIAL NAÏVE BAYES UNTUK ANALISIS SENTIMEN PELANGGGAN TOKOPEDIA,” J.Gauss, vol. 11, no. 4, pp. 562–571, Feb. 2023, doi: 10.14710/j.gauss.11.4.562-571.

[12] A. Nisa and E. Darwiyanto, “Analisis Sentimen Menggunakan Naive Bayes Classifier dengan Chi-Square Feature Selection Terhadap Penyedia Layanan Telekomunikasi”.

[13] Ramanda Md, Reza Dwi Restiyan, and Hafidz Irsyad, “Analisis Sentimen Masyarakat terhadap Perilaku Lawan Arah yang Diunggah pada Media Sosial Youtube Menggunakan Naïve Bayes,” BANDWIDTH, vol. 2, no. 2, pp. 75–83, Jul. 2024, doi: 10.53769/bandwidth.v2i2.706.

[14] R. A. Nurdian, Mujib Ridwan, and Ahmad Yusuf, “Komparasi Metode SMOTE dan ADASYN dalam Meningkatkan Performa Klasifikasi Herregistrasi Mahasiswa Baru,” JuTISI, vol. 8, no. 1, Apr. 2022, doi: 10.28932/jutisi.v8i1.4004.