Journal of Information Systems and Informatics
Vol. 7, No. 3, September 2025 e-ISSN: **2656-4882** p-ISSN: **2656-5935**
DOI: 10.51519/journalisi.v7i3.1196 Published By DRPM-UBD

Performance Comparison of Sentiment Classification Algorithms on SIGNAL Reviews Using SMOTE

Qothrunnada Wafi Anadia[1], Allsela Meiriza[2*]

[1,2] Information System, Faculty of Computer Science, Sriwijaya University, Indonesia
Email: [1]09031182227004@student.unsri.ac.id, [2*]allsela@unsri.ac.id

Abstract

Public service apps like SIGNAL are widely used to provide public access to information and vehicle tax payments. However, diverse user reviews highlight the need to evaluate public perception through sentiment analysis. Selecting an appropriate classification algorithm is crucial to ensure accurate results, particularly when dealing with imbalanced review data. Therefore, this study examines the comparative performance of four algorithms Naïve Bayes, Random Forest, Decision Tree, and SVM in analyzing the sentiment of 36,000 user feedback obtained from Google Play Store. The dataset underwent preprocessing, feature extraction using TF-IDF, and class balancing using SMOTE. Model evaluation was conducted using accuracy, precision, recall, and F1-score. The findings indicated that Random Forest performed the best overall performance (accuracy 91.04%, F1-score 94.80%), followed by Naïve Bayes (accuracy 89.89%, F1-score 93.38%), SVM (accuracy 89.22%, F1-score 93.02%), and Decision Tree (accuracy 88.40%, F1-score 92.31%). These findings indicate that Random Forest is highly effective for balanced datasets, while SVM and Naïve Bayes offer competitive precision for applications prioritizing accuracy in positive class detection. The output of this study can be applied practically by developers and related institutions in optimizing public service applications and by applying Random Forest algorithm to gain actionable insights for optimizing features and aligning services more closely with user needs.

Keywords: Sentiment Analysis, User Reviews, Classification Algorithm, SMOTE, Text Classification

1. INTRODUCTION

Information technology has developed rapidly and serves a key purpose amid the advancement of digitalization [1]. the government leverages information technology to provide better access to public resource management, enhancing integrity, transparency, and information accessibility [2]. One significant implementation in this regard is the SIGNAL (Samsat Digital Nasional) application. SIGNAL is a digital application developed with the aim of facilitating and simplifying public access to motor vehicle tax services, improve the efficiency and transparency of tax administration, and deliver services to the community

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

through technology [3]. Although intended to improve efficiency in public services, user responses have been varied, as reflected in reviews on platforms such as the Google Play Store. Therefore, assessing public perception through sentiment analysis becomes essential to support the continuous advancement of this digital public service, aiming to provide the best experience and services aligned with user needs.

Several previous studies have conducted sentiment analysis on the SIGNAL application. For instance, a study applied the Naïve Bayes algorithm for three-class classification, achieving an accuracy of 91.64% [4]. Another study compared four SVM kernels and recorded the highest accuracy of 92.2% on the linear kernel [5]. In another study, a comparison of algorithms between SVM, Naïve Bayes, and KNN was applied to social media comments, with SVM showing the best performance at 88% accuracy [6]. A lexicon-based approach was also used to evaluate three SVM kernels, with the linear kernel again showing the best performance [7]. Additionally, the analysis of SIGNAL and DigiKorlantas app reviews was conducted using the Naïve Bayes algorithm and the N-Grams approach, achieving an accuracy of 81.09% based on data from Google Play and the App Store [8].

Although several studies have investigated sentiment analysis of the SIGNAL application, most focus solely on evaluating a single classification algorithm [4], [5], [7] or comparing several models [6] without addressing the challenge of imbalanced data distribution, commonly known as class imbalance. Class imbalance commonly occurs and becomes a challenge in sentiment analysis of user reviews, where one sentiment class (e.g., positive) dominates the others (negative or neutral). These conditions can lead to the classification model being biased towards the majority class, which diminishes its effectiveness in accurately recognizing instances from the minority class [9]. Additionally, no previous studies have been identified that explicitly utilize techniques for managing class imbalance, like the Synthetic Minority Over-sampling Technique (SMOTE), in the analysis of SIGNAL user reviews.

To address this challenge, this research utilizes SMOTE to equalize the class distribution in the training data for sentiment analysis of SIGNAL user reviews, prior to measuring the capabilities of several classification algorithms to uphold the integrity of the evaluation results. With SMOTE, minority class samples are generated by determining the position between existing minority examples and their closest neighbours, thereby enabling a more balanced learning process [10] This step of balancing the data is expected to facilitate more accurate and unbiased evaluations of model performance [11]. Several previous studies have shown that the use of SMOTE can significantly improve classification performance and reduce bias, particularly in the field of sentiment analysis [12], [13], [14].

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

The selection of an appropriate classification algorithm is also crucial for obtaining optimal results in sentiment analysis. This study compares four classification algorithms selected based on their effectiveness and popularity in text classification [15][16]. Naïve Bayes is favored for its simplicity and efficiency, Random Forest excels at handling complex patterns and reducing overfitting, Decision Tree offers interpretability, and SVM is effective in maximizing the margin between classes in high-dimensional space. This comparison aims to identify the best method for balanced sentiment classification of SIGNAL user reviews. The uniqueness of this research is found in the integration of SMOTE-based data balancing with a comparative evaluation of various classification algorithms to produce a more objective and accurate sentiment analysis system. Thus, this study is directed to facilitate improvements of digital public services that are fairer, more adaptive, and data-driven.

2. METHODS

This research was conducted through eight systematic stages designed as guidelines or workflow steps to ensure the research process runs efficiently. These stages include collecting the data, labelling, pre-process, feature extraction, splitting the data, SMOTE oversampling, model implementation, and evaluation. the research workflow is shown in Figure 2.

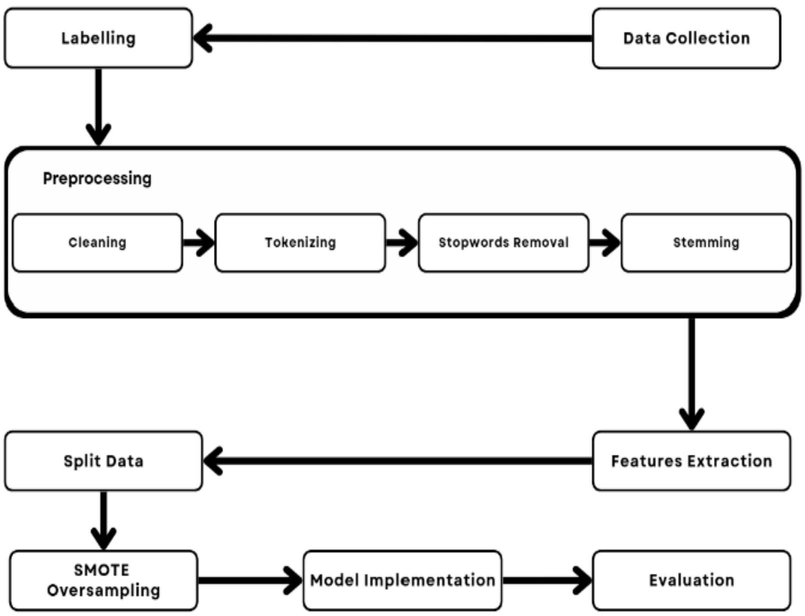

Figure 1. Research Stage

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

2.1. Data Collection

Data was collected using scraping techniques with the Google Play Scraper library on Google Colaboratory to obtain feedback from SIGNAL users available on the Google Play Store. This process enables the automatic retrieval of user review data, including information such as comment text, star ratings, and review upload time.

2.2. Labelling

Data labelling was performed automatically using a rating-based approach. Reviews assigned 4 or 5 stars were categorized as positive sentiment, whereas the rated 1 to 3 stars were categorized as negative sentiment [5], [17]. This approach was chosen due to its efficiency and relevance for application review data, as it utilizes the inherent attributes of the review data.

2.3. Preprocessing

The preprocessing stage aims to improve data quality before analysis. This process includes cleaning the text from unnecessary characters or symbols, tokenization to break sentences into single words, while removing the stopword is performed to discard commonly used words that are less significant, and stemming converts words into their fundamental form, a necessary step to structure the data and prepare it for classification model analysis [18].

2.4. Feature Extraction

The Term Frequency-Inverse Document Frequency (TF-IDF) technique is utilized to converting text-based information into numerical values suitable for machine learning models [19]. Term Frequency (TF) assessing the presence of words in a document based on their frequency, while Inverse Document Frequency (IDF) measuring the scarcity of word occurrence across all documents. This weighting helps highlight important terms while down-weighting common ones. The application of TF-IDF is prevalent in the field of text classification, clustering, and document similarity tasks [20]. The formula as follows:

$$\mathrm{TF}(t,d)=\frac{\text{Number of occurrences of term t in document d}}{\text{Total number of terms in the document d}} \tag{1}$$

$$\mathrm{IDF}(t)=\log\left(\frac{N}{\mathrm{DF}(t)}\right) \tag{2}$$

Description:
N represents documents total number, while DF(t) denotes documents number of t term appearance

Journal of Information Systems and Informatics
Vol. 7, No. 3, September 2025

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

In this study, the TF-IDF vectorizer was configured to extract features using a unigram and bigram range (n-gram range = 1 to 2). This configuration captures both individual words and common two-word phrases to better represent the context of user reviews. Standard stopword removal was applied to eliminate common, non-informative words (e.g., "dan," "yang," "adalah") which does not give significant value to the task of sentiment classification.

Prior to TF-IDF vectorization, the text data underwent preprocessing steps including tokenization, stemming, and removal of punctuation and special characters to ensure cleaner input. The resulting TF-IDF matrix was used as input for classification models, with feature importance analyzed through average TF-IDF scores to identify the most significant terms.

2.5. Split Data

The labelled dataset underwent a division into two subsets: for training is 80% allocated and testing is 20%. The split was shown randomly while maintaining the class distribution between positive and negative data to ensure balance in both subsets. This division is important to ensure that the model is not only capable of recognizing the data it was trained on but can also generalize to new, unseen data.

2.6. SMOTE Oversampling

To provide a clearer understanding of the SMOTE process, its main steps are illustrated in the following flowchart.

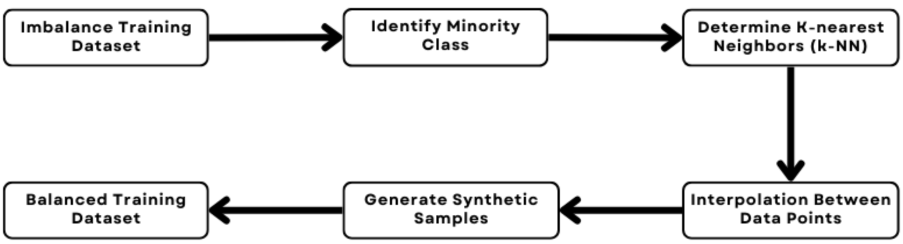

Figure 2. Flowchart of SMOTE

SMOTE generates synthetic data for minority classes in an effort to address class imbalance using interpolation among the closest data points [21]. In this study, the process was configured with k = 5 nearest neighbors to produce representative samples without introducing excessive noise. The quality of the synthetic data generated by SMOTE can be validated by evaluating the model on the original test data (without SMOTE) to ensure that the measured performance truly reflects the model's capability on real-world data [22].

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

SMOTE was applied only to the training data after the data splitting process to balance the training set, allowing the model to be more sensitive in learning patterns from all classes fairly [23]. This approach ensures a more effective training process and valid model evaluation, as the test data remains original and is not subjected to oversampling. Therefore, the implementation of SMOTE in this study also contributes to improving the capability of the model in classifying minority class more accurately and objectively.

2.7. Model Implementation

This study implemented four text classification algorithms, namely Naïve Bayes, SVM, Random Forest, and Decision Tree. All models are applied to the SIGNAL application review dataset, which has undergone labeling, preprocessing, and data balancing using SMOTE. Each model was implemented in Google Colaboratory using the Scikit-learn library, which also provided default hyperparameters, without additional hyperparameter adjustments. This approach was used to ensure fair results, as the use of default configurations allowed for consistent comparisons and could be directly replicated for another research.

2.8. Evaluation

Evaluation was carried out by the data that has been tested for performing each classification algorithm, with the goal of measuring the work of its models for divining or classifying unseen data was the main focus of this evaluation. The Confusion Matrix method was utilized to analyze both accurate and inaccurate predictions generated by the classification algorithms. The used of four key evaluation metrics: F1-Score, Recall, Precision, and Accuracy. Accuracy reflects the right number that have been predicted for all the number has been. Precision assesses the percentage of instances detected as positive, which is indeed a positive case. Recall measures the success rate of the model in identifying all true positive cases. F1-Score is a metric that calculated by combining precision and recall to assess model performance through an averaging method, resulting in a balanced assessment of both metrics. Each of the four metrics is determined through the confusion matrix. The results of this evaluation were used to determine which model is the most effective in the context of sentiment classification on SIGNAL application reviews. The four key evaluation metrics are determined using the Equation 3 to Equation 6.

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FN+FP}=\frac{TP+TN}{P+N} \tag{3}$$

$$\text{Precision}=\frac{TP}{TP+FP} \tag{4}$$

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

$$\text{Recall}=\frac{TP+TN}{TP+FN} \tag{5}$$

$$\text{F1-Score}=2 \times \frac{\text{Precision x Recall}}{\text{Precision+Recall}} \tag{6}$$

Description:

TP (True Positive) : Positive data that are correctly classified as positive.
TN (True Negative) : Negative data that are correctly classified as negative.
FP (False Positive) : Negative data that is mistakenly classified as positive.
FN (False Negative) : Positive data that is mistakenly classified as negative.
P : Total positive data (TP + FN)
N : Total negative data (TN + FP)

3. RESULTS AND DISCUSSION

This section is presenting the research output obtained by looking the methodology that has been describe above, starting from data collection to the model evaluation process.

3.1. Data Collection

This study collected 36,000 Reviews of the SIGNAL app. These results were obtained from Google Play Store through data collection based on scraping using the Google Play Scraper library. The review data was gathered within the time period from January 6, 2024, to May 31, 2025.

3.2. Labelling

This study uses a rating-based approach to perform automatic data labelling. Reviews rated 4 or 5 stars were labeled as positive sentiment, whereas those rated 1 to 3 stars were labeled as negative sentiment. The outcome of the labeling process is presented in Table 1.

Table 1. Labelling Sample

Content	Content (English)	Score	Label
Keren, semakin mudah pembayaran pajak motor. Tahun lalu coba bayar online tapi ambil bukti pajak di	Cool, paying motor vehicle tax is getting easier. Last year I tried paying online but still had to collect the tax receipt	5	Positive

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

Content	Content (English)	Score	Label
Samsat. Tahun ini coba kirim kerumah. Efisiensi waktu banget	at Samsat. This year I tried having it delivered to my home. Really time-efficient		
prosesnya penerbitan dan pengirimannya masih lama 1 bulan lebih	the issuance and delivery process still takes more than 1 month	3	Negative
sangat membantu.	very helpful.	4	Positive
tolong pemerintah d tindak lanjuti dengan aplikasi yang di buat tidak bisa di gunakan	please, government, follow up on this because the application created cannot be used	1	Negative

3.3. Preprocessing

The preprocessing stage produced text data that had been cleaned of irrelevant characters through a cleaning process, separated into individual words through tokenization, and stripped of common words with insignificant meaning through stopwords removal. Subsequently, each word was reduced to its root form through stemming. This process resulted in more structured and consistent text data, making it ready for feature extraction and training of the sentiment classification model. The output from preprocessing is shown in Table 2.

Table 2. Preprocessing

Preprocessing	Content	Content (English)
Original Text	"Keren, semakin mudah pembayaran pajak motor. Tahun lalu coba bayar online tapi ambil bukti pajak di Samsat. Tahun ini coba kirim kerumah. Efisiensi waktu banget"	"Cool, paying motorbike tax is getting easier. Last year I tried paying online but still had to collect the tax proof at the Samsat office. This year I tried having it delivered to my house. It really saves time."
Cleaning	"keren semakin mudah pembayaran pajak motor tahun lalu coba bayar online tapi ambil bukti pajak di samsat tahun ini coba kirim	"cool paying motorbike tax is getting easier last year tried to pay online but had to collect the tax proof at samsat this year tried to have it delivered

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

Preprocessing	Content	Content (English)
	kerumah efisiensi waktu banget"	home really time efficient"
Tokenizing	['keren', 'semakin', 'mudah', 'pembayaran', 'pajak', 'motor', 'tahun', 'lalu', 'coba', 'bayar', 'online', 'tapi', 'ambil', 'bukti', 'pajak', 'di', 'samsat', 'tahun', 'ini', 'coba', 'kirim', 'kerumah', 'efisiensi', 'waktu', 'banget']	['cool', 'getting', 'easier', 'paying', 'motorbike', 'tax', 'last', 'year', 'tried', 'to', 'pay', 'online', 'but', 'collect', 'tax', 'proof', 'at', 'samsat', 'this', 'year', 'tried', 'to', 'deliver', 'to', 'home', 'really', 'time', 'efficient']
Stopwords Removal	['keren', 'mudah', 'pembayaran', 'pajak', 'motor', 'coba', 'bayar', 'online', 'ambil', 'bukti', 'pajak', 'samsat', 'coba', 'kirim', 'kerumah', 'efisiensi', 'banget']	['cool', 'easier', 'paying', 'motorbike', 'tax', 'tried', 'pay', 'online', 'collect', 'tax', 'proof', 'samsat', 'tried', 'deliver', 'home', 'time', 'efficient']
Stemming	['keren', 'mudah', 'bayar', 'pajak', 'motor', 'coba', 'bayar', 'online', 'ambil', 'bukti', 'pajak', 'samsat', 'coba', 'kirim', 'rumah', 'efisiensi', 'banget']	['cool', 'easy', 'pay', 'tax', 'motorbike', 'try', 'pay', 'online', 'collect', 'tax', 'proof', 'samsat', 'try', 'deliver', 'home', 'efficient']

3.4. Features Extraction

In this stage, Feature extraction was carried out through the application of the TF-IDF (Term Frequency–Inverse Document Frequency) method. This technique was applied to encode the stemmed words as numerical values, indicating their importance in a given document relative to the whole corpus. The output of the feature extraction process is presented in Table 3.

Table 3. Features Extraction

Word	TF-IDF
bantu	0.077499
mudah	0.065678

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Word	TF-IDF
mantap	0.050902
cepat	0.049287
bayar	0.042447
pajak	0.036251
aplikasi	0.033665
kirim	0.030160
proses	0.030160
bagus	0.028597

A high TF-IDF value suggests that a word is common in a specific review while being infrequent in other reviews, and is therefore considered more informative. Based on the extraction results, words such as "bantu" (0.077), "mudah" (0.066), and "mantap" (0.051) had the highest TF-IDF scores. This indicates that these words strongly contribute to representing user sentiment toward the SIGNAL application.

3.5. Split Data

The data collection is grouped into two subsets: training and testing sets. A total of 36,000 data entries were used, which were split using an 80:20 ratio, resulting in 28,800 training data and 7,200 testing data.

3.6. SMOTE Oversampling

To balance the training data, the Synthetic Minority Over-sampling Technique (SMOTE) was implemented to improve the uneven distribution of classes between positive and negative sentiment. The result of this stage was a training dataset with balanced class proportions. The results of SMOTE implementation is shown in Figure 3.

Journal of Information Systems and Informatics
Vol. 7, No. 3, September 2025
p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

Figure 3. SMOTE Oversampling

3.7. Model Implementation

In this stage, four classification algorithms were tested: Support Vector Machine (SVM), Naïve Bayes, Decision Tree, and Random Forest to compare their performance in analyzing sentiment from SIGNAL application reviews.

1) Support Vector Machine (SVM)

The confusion matrix generated by the SVM model is shown in Figure 4. The confusion matrix illustrates the ability of the SVM model to classify the majority of the data correctly, with 1,267 negative data and 5,161 positive data classified accurately, while misclassifying 137 and 635 data, respectively.

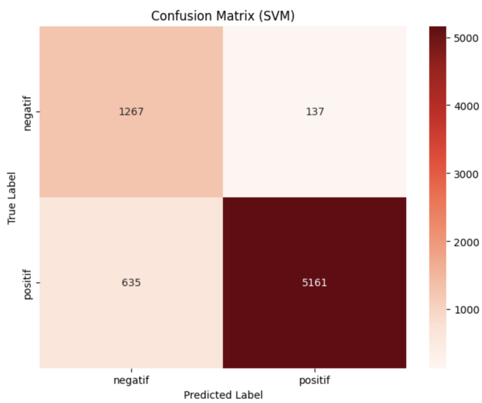

Figure 4. Confusion Matrix SVM

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2) Naïve Bayes

The confusion matrix generated by the Naïve Bayes model is shown in Figure 5. The confusion matrix illustrates the ability of the Naïve Bayes model to classify the majority of the data correctly, with 1,268 negative data and 5,204 positive data classified accurately, while misclassifying 136 and 592 data, respectively.

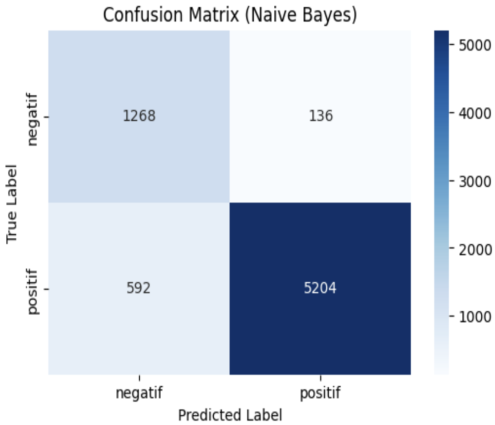

Figure 5. Confusion Matrix Naïve Bayes

3) Random Forest

The confusion matrix generated by the Random Forest model is shown in Figure 6. The confusion matrix illustrates the ability of the Random Forest model to classify the majority of the data, with 1,202 negative data and 5,353 positive data accurately predicted, while 202 and 443 data were misclassified, respectively.

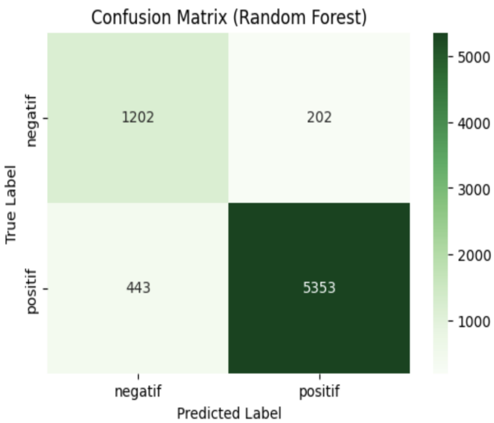

Figure 6. Confusion Matrix Random Forest

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

4) Decision Tree

The confusion matrix generated by the Decision Tree model is shown in Figure 7. The confusion matrix illustrates the ability of the Decision Tree model to classify the majority of the data, with 1,057 negative data and 5,310 positive data accurately predicted, while 347 and 486 data were misclassified, respectively.

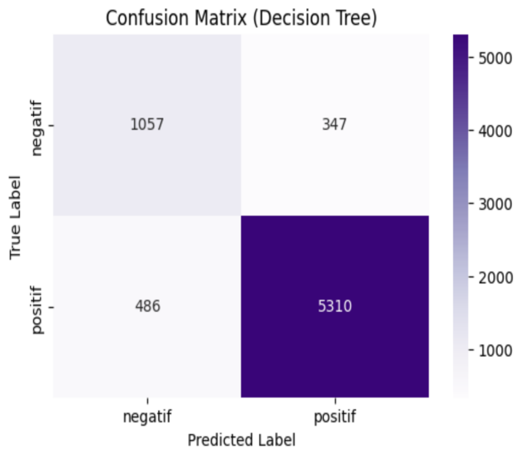

Figure 7. Confusion Matrix Decision Tree

3.8. Model Evaluation

The performance of each classification model was assessed and compared through accuracy, precision, recall, and F1-score values derived from the confusion matrix, and the results are presented in Table 4.

Table 4. Model Evaluation

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	91.04%	96.36%	92.35%	94.80%
Naive Bayes	89.89%	97.45%	89.76%	93.38%
SVM	89.22%	97.41%	89.04%	93.02%
Decision Tree	88.40%	93.86%	91.61%	92.31%

Table 4 presents a performance summary for each of the four tested algorithms, namely SVM, Naïve Bayes, Random Forest, and Decision Tree, with evaluation metrics in the form of accuracy, precision, recall, and F1-score. Overall, the most prominent model in this test was Random Forest, recording an accuracy of

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

91.04%, the highest recall of 92.35%, and an F1-score of 94.80%. Although Naïve Bayes (97.45%) and SVM (97.41%) are slightly higher than Random Forest (96.36%) in terms of precision, this difference reflects how each algorithm handles classification errors. The high precision of Naïve Bayes and SVM indicates their superior ability to minimise false positives, which is important when positive prediction errors have significant consequences. However, the relatively lower recall compared to Random Forest indicates a trade-off, as some positive cases are not detected. The combination of high precision and recall in Random Forest results in the highest F1-score, confirming the algorithm's effectiveness in reducing both false positives and false negatives.

3.9. Discussion

The evaluation results highlight notable differences in the performance of the four algorithms. These variations can be better understood by examining the characteristics of the SIGNAL user review dataset and the underlying mechanisms of each algorithm. As an ensemble method, Random Forest works by combining several decision trees, using a majority voting mechanism to obtain the final result, tends to better capture complex patterns and interactions in imbalanced and noisy text data compared to single-model approaches. Its resilience to overfitting allows it to generalize more effectively even when the data is highly complex [24], [25]. Meanwhile, Naïve Bayes, which relies on probabilistic assumptions, and SVM, which focuses on maximizing the margin between classes, often achieve high precision but sometimes at the cost of recall due to their sensitivity to minority-class patterns. Decision Tree, which scored lower, demonstrates limitations in capturing complex patterns without the support of ensemble techniques.

When selecting an algorithm for an application, assesing the balance between precision, recall, and F1-score is crucial. If the priority is to minimize false positive, Naïve Bayes or SVM may be preferable. Conversely, if the focus is on capturing as many positive cases as possible, Random Forest with its higher recall and F1-score becomes the more appropriate choice. F1-score serves as a comprehensive metric to assess this trade-off, enabling decisions to be tailored to the specific priorities of the application.

Nonetheless, several limitations should be noted. Although Random Forest is relatively resistant to overfitting, excessive complexity or uneven feature distribution can affect its generalization capability on new data. Furthermore, while SMOTE effectively balances the training data by creating synthetic samples that represent minority classes, it also has the potential to introduce noise or overlap between classes, which may impact model robustness.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

4. CONCLUSION

In this study, the effectiveness of four classification models SVM, Naïve Bayes, Random Forest, and Decision Tree was evaluated in analyzing user sentiment regarding the SIGNAL application. The analysis involved several stages, including collecting user review data, automatically labeling the data based on star ratings, text preprocessing, feature extraction using the TF-IDF method, and handling class imbalance with SMOTE. Among the models tested, Random Forest recorded excellent performance with an accuracy of 91.04% and an F1 score of 94.80%. These findings indicate that applying balancing techniques such as the involvement of SMOTE in ensuring a balanced data distribution, thereby enabling a more valid model evaluation process and facilitating the selection of the most optimal-performing model.

The insights gained from this study can be applied to other domains with similar sentiment analysis tasks, such as product reviews or customer feedback systems, where class imbalance issues are also common. For public service administrators, the approach used in this study can also be applied to other public service sentiment analysis systems, such as e-government portals or citizen complaint platforms. Combining the right classification algorithms with data balancing techniques can help improve analysis accuracy and more effectively support service improvements.

This study highlights certain limitations, such as the potential for overfitting, particularly in ensemble models such as Random Forest. Therefore, validating the model on different and larger datasets is strongly recommended to ensure generalization capability. For the next research, it is recommended to test the models on other datasets and explore the utilization of additional features, such as sentiment lexicons or deep learning–based embedding representations, with the aim of improving the accuracy and effectiveness of the model as a whole.

REFERENCES

[1] C. S. Octiva, P. E. Haes, T. I. Fajri, H. Eldo, and M. L. Hakim, "Implementasi teknologi informasi pada UMKM: Tantangan dan peluang," *J. Minfo Polgan*, vol. 13, no. 1, pp. 815–821, Jul. 2024.
[2] C. Mauni, "Peran teknologi informasi dalam meningkatkan transparansi dan akuntabilitas pemerintah daerah Jawa Timur," *J. Ilmu Sosial dan Humaniora (ISORA)*, vol. 3, no. 1, pp. 104–115, Jan.–Jun. 2025.
[3] D. Kurniadi and T. Sutabri, "Penerapan aplikasi Samsat Digital sebagai layanan publik di Sumatera Selatan," *Switch: Jurnal Sains dan Teknologi Informasi*, vol. 3, no. 1, pp. 49–56, Dec. 2024.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

[4] R. Chandra and E. M. Sipayung, "Analisis Sentimen Ulasan Aplikasi Samsat Digital Nasional Menggunakan Algoritma Naive Bayes Classifier," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 10, no. 3, pp. 156–164, Jan. 2025.

[5] N. Y. Setyawati and E. Widarti, "Analisis sentimen aplikasi Samsat Digital di Play Store menggunakan Support Vector Machine," *J. Smart Syst.*, vol. 3, no. 2, pp. 33–42, Jan. 2024.

[6] A. S. Kirana, R. Rusdah, R. Roeswidiah, and A. Pudoli, "Analisis sentimen pada media sosial terhadap layanan Samsat Digital Nasional dengan Support Vector Machine," *Idealis: Indonesia Journal Information System*, vol. 8, no. 1, pp. 53–63, Jan. 2025.

[7] S. Kacung, C. P. P. Bagyana, and D. Cahyono, "Analisis sentimen terhadap layanan Samsat Digital Nasional (SIGNAL) menggunakan metode SVM," *J. MNEMONIC*, vol. 7, no. 1, pp. 118–122, Feb. 2024.

[8] H. M. Puspasari, I. Z. Mustaqim, A. T. Utami, R. Syalevi, and Y. Ruldeviyani, "Evaluation of Indonesia's police public service platforms through sentiment and thematic analysis," *IAES International Journal of Artificial Intelligence*, vol. 13, no. 2, pp. 1596–1607, Jun. 2024.

[9] A. Syukron, E. Saputro, and P. Widodo, "Penerapan Metode Smote Untuk Mengatasi Ketidakseimbangan Kelas Pada Prediksi Gagal Jantung," *J-TIT: j. Teknologi Informasi dan Terapan,* Vol. 10 No. 1 Juni 2023

[10] C. Candra, K. W. Chandra, and H. Irsyad, "Efektifitas SMOTE dalam Mengatasi Imbalanced Class Algoritma K-Nearest Neighbors pada Analisis Sentimen terhadap Starlink," *Jurnal Ilmu Komputer dan Informatika*, vol. 4, no. 1, pp. 31–42, Jul. 2024.

[11] A. Tripathi, R. Chakraborty, and S. K. Kopparapu, "A Novel Adaptive Minority Oversampling Technique for Improved Classification in Data Imbalanced Scenarios," *arXiv preprint arXiv:2103.13823*, 2021.

[12] F. Dwianasari, R. D. Yani, K. N. Laksono, N. Mujaliza, and R. Fahlapi, "Analisis Sentimen Masyarakat terhadap Aktivitas Pertambangan di Raja Ampat Menggunakan Support Vector Machine dan Naïve Bayes dengan Teknik SMOTE," *Kajian Ekonomi dan Akuntansi Terapan*, vol. 2, no. 2, pp. 234-244, Juni 2025

[13] A. H. Luthfi, A. Faqih, and G. Dwilestari, "Enhancing Model Accuracy in Sentiment Analysis of the by.U Application Using Naïve Bayes and SMOTE Techniques," *J. of Artificial Intelligence and Engineering Applications,* vol. 4, no. 2, Feb 2025.

[14] F. Destiyanti, A. I. Hadiana, and F. R. Umbara, "Penerapan Metode Support Vector Machine dan SMOTE untuk Klasifikasi Sentimen Publik Terhadap Polisi Republik Indonesia," *JUMANJI*, vol. 8, no. 1, pp. 1-15, April 2024.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

[15] M. F. Naufal, T. Arifin, and H. Wirjawan, "Analisis perbandingan tingkat performa algoritma SVM, Random Forest, dan Naïve Bayes untuk klasifikasi cyberbullying pada media sosial," *JURASIK: Jurnal Riset Sistem Informasi dan Teknik Informatika*, vol. 8, no. 1, pp. 70–81, Feb. 2023.
[16] O. S. D. Fadhillah, J. H. Jaman, and C. Carudin, "Perbandingan Naive Bayes, Support Vector Machine, Logistic Regression dan Random Forest dalam menganalisis sentimen mengenai TikTokshop," *J. Inform. dan Tek. Elektro Terapan*, vol. 13, no. 1, Jan. 2025.
[17] S. A. H. Bahtiar, C. K. Dewa, and A. Luthfi, "Comparison of Naïve Bayes and Logistic Regression in Sentiment Analysis on Marketplace Reviews Using Rating-Based Labeling," *Journal of Information Systems and Informatics*, vol. 5, no. 3, pp. 915–927, Aug. 2023.
[18] S. R. Putri, M. Arifin, and Supriyono, "Analisis Sentimen Publik terhadap Nadiem Makarim sebagai Mendikbudrisktek menggunakan Support Vector Machine (SVM)," *Sistemasi: Jurnal Sistem Informasi*, vol. 14, no. 2, pp. 826-834, 2025.
[19] T. Dodiya, "Using Term Frequency - Inverse Document Frequency to find the Relevance of Words in Gujarati Language," *Int J Res Appl Sci Eng Technol*, vol. 9, no. 4, pp. 378–381, Apr. 2021.
[20] H. Sari, G. L. Ginting, T. Zebua, and Mesran, "Penerapan Algoritma Text Mining dan TF-IDF untuk Pengelompokan Topik Skripsi pada Aplikasi Repository STMIK Budi Darma," *TIN: Terapan Informatika Nusantara*, vol. 2, no. 7, pp. 414–432, Dec. 2021.
[21] F. Sidik and I. F. Mahdy, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Sentimen Genosida Palestina Menggunakan Multinomial Naive Bayes," *Bandung Conference Series Statistics*, vol. 4, no. 2, pp. 320-328, Aug. 2024.
[22] E. Espinosa and A. Figueira, "On the quality of synthetic generated tabular data," *Mathematics*, vol. 11, no. 15, Art. no. 3278, Jul. 2023, doi: 10.3390/math11153278.
[23] R. A. Danquah, "Handling Imbalanced Data: A Case Study for Binary Class Problems," *arXiv preprint arXiv:2010.04326*, 2020.
[24] A. K. Hidayah, Y. Erwadi, and S. Handayani, "Analisis Sentimen Publik Terhadap Pemindahan Ibu Kota Negara di Twitter Menggunakan Metode Klasifikasi Random Forest dan SMOTE," *JATI*, vol. 9, no. 5, pp. 9109-9114, Oct. 2025.
[25] I. Muhandhis, A. S. Ritonga, W. Putra, and R. Benowo, "Public sentiment analysis on TikTok about Tapera policy using Random Forest classifier," *Sistemasi: J. Sistem Informasi*, vol. 14, no. 1, pp. 354–365, 2025.