Journal of Applied Informatics and Computing (JAIC)
Vol.9, No.6, December 2025, pp. 3697~3705
e-ISSN: 2548-6861

An Ensemble Learning Approach for Sentiment Analysis of Maxim Application Reviews Using SVM, KNN, and Random Forest

Ruth Mei Sasmita [1]*, Allsela Meiriza [2]*, Hardini Novianti [3]*
* Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Sriwijaya
09031182227008@student.unsri.ac.id [1], allsela@unsri.ac.id [2], hardininovianti@gmail.com [3]

Article Info

Article history:

Received 2025-10-07
Revised 2025-10-30
Accepted 2025-11-05

Keyword:

Ensemble Learning,
K-Nearest Neighbor,
Random Forest,
Sentiment Analysis,
Support Vector Machine.

ABSTRACT

The development of online transportation applications such as Maxim has increased the need for sentiment analysis to understand user opinions from reviews on the Google Play Store. The main challenges in this analysis are language diversity, variations in writing style, and data imbalance, which affect model accuracy. This study aims to evaluate the performance of the Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Random Forest (RF) algorithms, as well as ensemble approaches through the Voting Classifier and Combined Classifier, in sentiment analysis of Maxim app reviews. The dataset consists of 2,851 Indonesian-language reviews collected through web scraping from the Google Play Store in 2025. Sentiment labels were automatically determined based on user ratings, where ratings of 4–5 were categorized as positive and ratings below 4 as negative, with an initial distribution of 2,295 positive and 556 negative reviews before balancing using SMOTE–Tomek Links. Preprocessing steps included case folding, tokenization, stopword removal, and stemming using Sastrawi, while feature weighting was performed with unigram TF-IDF. The Combined Classifier merged the probability scores from the SVM, KNN, and RF models to produce the final prediction. Evaluation was conducted using 5-Fold Cross Validation with accuracy, precision, recall, F1-score, and ROC-AUC as evaluation metrics. The results show that RF and the Combined Classifier achieved the best performance with 85% accuracy, 87% precision, 85% recall, 86% F1-score, and 0.91 ROC-AUC, while SVM and the Voting Classifier ranked in the middle and KNN ranked the lowest. These findings confirm that ensemble learning, particularly the Combined Classifier, effectively improves the accuracy and stability of review classification compared to individual methods.

I. INTRODUCTION

The development of online transportation applications in Indonesia, such as Maxim, has facilitated people's mobility and has become increasingly in demand across various cities. User reviews of Maxim on the Google Play Store serve as an important source of data for evaluating service quality. However, sentiment analysis of these reviews faces several challenges, including language diversity, variations in writing styles, and data imbalance, which can result in unstable classification accuracy [1]. The ensemble learning approach has emerged as one of the most widely researched solutions for improving the robustness and accuracy of text classification models. Ensemble learning combines predictions from multiple base learners, allowing model-specific errors to be offset, thereby resulting in more stable and often superior performance compared to single models across various datasets and domains. Previous research has also reinforced this, showing that ensemble approaches such as Random Forest and Boosting can predict sentiment more effectively than individual algorithms [2]. Research by [3] indicates that the application of a Voting Classifier, which merges different algorithms such as Logistic Regression and Random Forest, contributes to improved sentiment analysis

outcomes. Another study [4] shows that both heterogeneous ensemble techniques (combining different models) and homogeneous ensemble techniques (using multiple similar models) can improve accuracy and generalization in sentiment analysis tasks compared to single-model approaches.

This work implemented three established machine learning algorithms that are widely recognized in text classification, namely Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Random Forest (RF). Single-model approaches, particularly those utilizing SVM and KNN, have been extensively applied in sentiment analysis literature. Specifically, SVM demonstrates exceptional performance when dealing with high-dimensional text features [4]. This is supported by research [5] on Shopee sentiment analysis, which demonstrates that SVM can achieve 98% accuracy in distinguishing between positive and negative comments. Meanwhile, KNN is capable of classifying data based on pattern proximity [6], and with proper hyperparameter tuning, it can reach an accuracy of up to 98.37% in multilingual Twitter sentiment analysis [7]. Similarly, Random Forest, as an ensemble method that is robust against noise and overfitting [8], has shown superior performance, achieving 99.88% accuracy, 99.88% recall, 99.93% precision, and a 99.88% F1-score in predicting pressure ulcers [9]. These findings indicate that RF is highly effective in achieving reliable and high accuracy.

However, research indicates that single methods often yield limited accuracy when addressing language variations and imbalanced data. For example, a study on Maxim app reviews reported that SVM achieved only 79% accuracy [10], while another study recorded higher accuracy at 96% [11]. Therefore, the ensemble learning approach, including Voting and Combined Classifiers, is considered more capable of handling challenges related to complexity and variation in classification objects [12], as demonstrated in research on fake review detection [13]. The performance of ensemble learning was examined in this research by applying K-Fold Cross Validation. Evaluation metrics included accuracy, precision, recall, and F1-score, which were utilized to determine the optimal approach relative to standalone algorithms. Taking these considerations into account, the present study proposes and evaluates an ensemble learning strategy for sentiment analysis of Maxim reviews collected from the Google Play Store, where SVM, KNN, and Random Forest serve as the base learning models. In addition, this work compares individual algorithms (SVM, KNN, RF) with ensemble approaches, specifically majority voting and the combined classifier. The Maxim platform was selected as the focal case study owing to the relatively limited scholarly investigations on this service compared with competitors such as Gojek and Grab, and the heterogeneity of its user feedback, which creates both challenges and prospects for assessing the utility of ensemble learning.

This study contributes to advancing ensemble-based sentiment analysis methods by improving the accuracy and consistency of review classification in online transportation applications. Furthermore, the findings of this study can serve as a reference for future research in text mining and machine learning, while also providing service providers with a more representative understanding of user opinions to support data-driven decision-making.

II. METHOD

This study constructs an ensemble model using SVM, KNN, and RF as base learners for sentiment analysis of Maxim reviews. The process is carried out systematically through several stages, including data collection, preprocessing, balancing, dataset partitioning, feature weighting, base model training, and integration into the ensemble. The overall research flow is illustrated in Figure 1.

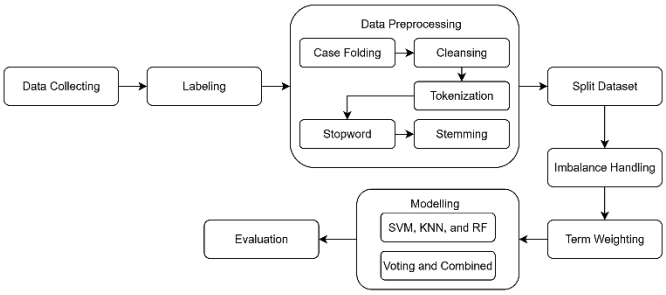

Figure 1. Research Methodology

A. Data Collecting

Data collecting is defined as an automated technique for extracting text data from social media [14]. For this study, user reviews of the Maxim application from the Google Play Store were collected as the dataset. The dataset comprises reviews from Indonesian users collected in 2025. The acquisition process employed a scraping method utilizing the Python package *google-play-scraper*.

B. Labeling

In the context of data mining and machine learning, labeling refers to the process of assigning labels to raw data so that it can be used as the target output during model training [14]. The weighting of words is determined by their frequency of occurrence in a document, where increased repetition reflects a higher degree of importance in the context of the text [15].

C. Preprocessing Data

Text preprocessing is the initial stage aimed at removing or minimizing noise in text data to prepare it for further processing [14]. The preprocessing procedure consists of case folding, cleansing, tokenization, stopword removal, and stemming. Implementing this pipeline has demonstrated notable improvements in the accuracy of Indonesian text classification models. The stopword removal and stemming

processes were performed using the Sastrawi library, a rule-based Indonesian language stemmer designed to reduce morphological variations in words.

D. Split Dataset

Subsequently, the dataset was partitioned into training and testing portions, facilitating an objective and independent assessment of the model's performance. The purpose of this division is to assess model performance using data that were not involved in training. In this study, an 80:20 ratio was applied, following standard practice in machine learning research, to ensure a balanced proportion between training and evaluation data [15].

E. Imbalance Handling

To address class imbalance between majority and minority reviews, the *SMOTE–Tomek Links* technique was applied. This method is effective in improving the performance of classification models, particularly in the context of healthcare reviews [16].

F. Term Weighting

Term weighting is the process of assigning numerical values to words in a document to represent their importance both within the document and across the corpus. The Term Frequency–Inverse Document Frequency (TF-IDF) unigram approach was utilized in this research as a means of calculating the weights assigned to individual words [17] [18].

$$\text{TF}= \frac{\text{number of occurrences of a word}}{number\ of\ words\ in\ a\ document} \quad (1)$$

$$IDF\ = log.\frac{number\ of\ documents}{number\ of\ occurrences\ of\ a\ word} \quad (2)$$

$$TFIDF = TF\ .IDF \quad (3)$$

G. Modelling

In the modeling process, three well-established machine learning algorithms were employed for text classification tasks, namely Support Vector Machine, K-Nearest Neighbor, and Random Forest. In addition, two ensemble learning approaches, Voting Classifier and Combined Classifier, were applied. The rationale for employing these algorithms and ensemble approaches lies in their distinct features and strengths, which are expected to contribute to higher accuracy and greater stability in classifying sentiments within Maxim app reviews.

1. Support Vector Machine

As a supervised machine learning method, Support Vector Machine (SVM) has proven highly effective for text classification, particularly in managing the complex, high-dimensional feature spaces generated through term-weighting approaches. The optimization objective of SVM is formulated as:

$$\min L(w,b) = \frac{1}{2}w^T w \quad (4)$$
$$\text{Subject to, } y_i(w^T x_i + b) \geq 1,\ \ i = 1, 2, \dots, n$$

2. K-Nearest Neighbor

In the K-Nearest Neighbor (KNN), classification of unseen instances is achieved by referencing the k most similar neighbors in the training set. The algorithm determines similarity using measures such as Euclidean, Cosine, or Manhattan distance, and assigns the instance to the majority class among those neighbors [19]. In the context of text data, such as sentiment analysis of Maxim app reviews, Cosine Similarity is often preferred because it effectively measures the similarity between high-dimensional sparse vectors :

$$CosSim(x,y) = \frac{\sum_{i=1}^{n} x_i\, y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}} \quad (5)$$

3. Random Forest

Random Forest is capable of handling large and complex datasets by generating accurate predictions and minimizing the risk of overfitting [20]. This algorithm is particularly suitable for sentiment analysis, as it can effectively capture complex relationships between review features and classification categories.

Random Forest formula:

$$f(x)\ \ = \frac{1}{N}\sum_{n=1}^{N} f_n(x) \quad (6)$$

Where $f_n(x)$ denotes the prediction generated by the *n*-th decision tree, while *N* indicates the total number of trees within the Random Forest.

4. Ensemble Classifier

In this study, an ensemble classifier was employed to enhance the accuracy and stability of sentiment analysis on Maxim app reviews. The ensemble approach integrates predictions from several base models (SVM, KNN, and Random Forest), allowing individual model errors to be compensated and thereby achieving better performance than a single model. This study employs an ensemble approach consisting of the Weighted Voting Classifier and the Combined Classifier. The Weighted Voting Classifier determines the final outcome based on the majority votes of the base models (SVM, KNN, and Random Forest), taking into account the performance weight of each model on the training data. In contrast, the Combined Classifier integrates the prediction probability scores (soft combination) of the three models to determine the final class. This approach differs from the Weighted Voting method as it considers the confidence level of each model rather than merely its predicted class, thereby producing more stable and reliable decisions, particularly when dealing with imbalanced data distributions. In a voting classifier, the final decision is determined by the majority of predictions from the base

learners (majority voting) [21]. In contrast, a combined classifier utilizes the aggregated probabilities or prediction scores from each model to determine the final class [12]. Both methods are effective in addressing data variation and class imbalance, as well as in improving model generalization.

H. Evaluation

Model performance evaluation was carried out to assess the effectiveness of each algorithm, including single models (SVM, KNN, and RF) and two ensemble approaches, namely the Weighted Voting Classifier and the Combined Classifier. A comparison between the Weighted Voting and Combined Classifiers was conducted to analyze the impact of the prediction combination mechanism on model stability and accuracy. The Weighted Voting Classifier was evaluated based on the dominance of the model with the highest assigned weight, whereas the Combined Classifier was assessed based on the averaged prediction probabilities to produce more consistent classification decisions.

Performance assessment of the model relied on widely adopted sentiment analysis indicators, namely accuracy, precision, recall, and F1-score. Complementary evaluation was conducted through confusion matrix analysis, which depicts correct and misclassified instances across categories, along with ROC curves and AUC values that assess the model's discriminative capability between positive and negative classes [22]. To enhance the validity of the results, K-Fold Cross-Validation was employed. The evaluation metrics used are as follows:

1. *Accuracy* is defined as the ratio between correctly classified observations both positive and negative and the total dataset size.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+F_x} \quad (7)$$

2. *Precision* is defined as the ratio of correctly identified positive cases to the entire set of cases labeled as positive by the model.

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

3. *Recall* is defined as the ratio between true positive predictions and the overall count of actual positive instances.

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

4. *F1-Score* is derived from the harmonic mean of precision and recall, delivering a comprehensive assessment that incorporates errors from both false positives and false negatives.

$$F1 = 2.\frac{precision\ .\ recall}{precision+recall} \quad (10)$$

Description:

1. *True Positive (TP)*: instances labeled as positive that are correctly classified as positive.
2. *True Negative (TN):* instances labeled as negative that are correctly classified as negative.
3. *False Positive (FP):* instances labeled as negative that are incorrectly classified as positive.
4. *False Negative (FN):* instances labeled as positive that are incorrectly classified as negative.

The model evaluation process was conducted on the Google Colaboratory platform using Python 3.10 with the *scikit-learn*, *pandas*, and *Sastrawi* libraries, with a random seed of 42 set to ensure result reproducibility. To minimize the risk of overfitting on a relatively small dataset, several mitigation strategies were applied. First, 5-Fold Cross Validation was employed as the primary technique to obtain a more stable model performance estimate and reduce bias caused by a single data split. Second, SMOTE–Tomek Links were utilized to reduce the complexity of decision boundaries by removing noise within the majority class. Third, the LinearSVC algorithm (as one of the base learners) was equipped with an L2 regularization mechanism to prevent the model from becoming overly complex. Finally, all base models (SVM, KNN, and Random Forest) were trained using the default hyperparameter settings commonly adopted in similar studies to avoid overfitting resulting from excessive parameter tuning on small datasets.

III. RESULTS AND DISCUSSION

This part of the study presents the results of applying several algorithms, including SVM, KNN, Random Forest, Ensemble Voting, and the Combined Classifier, for sentiment analysis of Maxim reviews sourced from the Google Play Store. The effectiveness of each model was analyzed through widely adopted evaluation measures, namely accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC.

A. Data Collecting

The dataset summarized in Table I comprises 2,851 reviews of the Maxim application, collected from the Google Play Store using web scraping techniques. The table includes information on usernames, review texts, ratings, and upload times.

TABLE I
MAXIM APPLICATION REVIEW DATA

Username	Review	Rating	At
Pengguna Google	mantap	5.0	05/08/2025 12:44
Pengguna Google	terbaik,sangat membantu sekali..	4.0	05/08/2025 11.35
Pengguna Google	semoga kedepannya	3.0	05/08/2025 11:18

	semakin baik apknya		
Pengguna Google	semoga lebih baik	5.0	05/08/202508:46
Kembalikan Sistem Pembagian Orderan seperti Tahun Sebelumnya..... ini pembagian orderan sangat Tidak logis	kembali sistem bagi order logis		

B. Labeling

The collected reviews were automatically labeled into sentiment categories based on rating rules. Reviews with ratings of 4–5 were classified as positive, whereas those with ratings below 4 were classified as negative. This labeling process resulted in an initial distribution of 2,295 positive reviews and 556 negative reviews. The distribution indicates a substantial class imbalance, necessitating the use of data balancing techniques in the subsequent stage to prevent model performance from being biased toward the majority class. The results of the automatic labeling process are presented in Table II.

TABLE II
LABELING REVIEWS BASED ON RATING

Content	Score	Label
RA TAU ONO ORDERAN....GATEL..	1.0	Negatif
terbaik,sangat membantu sekali..	5.0	Positif
Awal2 ada Titik hp customer(icon kepala orang) sangat akurat jadi mskpn titik jemput meleset TDK Masalah TPI akhir2 ini titik hp ikut2 an TDK akurat di maps ada di kiri kenyataannya di kanan bhkn kadang lmyn jauh jga, mohon segera diperbaiki seperti dlu lah minimal	2.0	Negatif
sejak ada driver Maxim kami bisa menyambung hidup sehari hari,dulu kami pengangguran sekarang udah bisa beraktivitas ojek	5.0	Positif

C. Data Preprocessing Result

The user review data of the Maxim application collected from the Google Play Store initially contained irrelevant elements such as capital letters, numbers, punctuation marks, emoticons, and common words. The text was normalized and prepared for feature extraction through preprocessing procedures including case folding, cleansing, tokenization, stopword removal, and stemming. The results of the preprocessing process are presented in Table III.

TABLE III
PREPROCESSING RESULT

Text	Clean_text
mantap	mantap
keren	keren
RA TAU ONO ORDERAN….GATEL..	ra tau ono order gatel
terbaik,sangat membantu sekali..	baik sangat bantu sekali
semoga kedepannya tambah baik apknya	moga depan tambah baik apknya

D. Split Dataset

The dataset was partitioned into training (80%, or 2,280 instances) and testing (20%, or 571 instances) prior to the balancing process. This allocation allowed the model to learn from a majority of the data while being evaluated on previously unseen cases. Such partitioning is fundamental for ensuring unbiased assessment and reflecting the model's effectiveness in real-world applications.

E. Imbalance Handling

The distribution of review data before and after balancing is presented in Table IV. Initially, there were 2,295 positive reviews and 556 negative reviews, indicating a substantial class imbalance. Such an imbalance may bias the model toward the majority class and reduce classification performance. To overcome the imbalance, the SMOTE–Tomek Links strategy was implemented. The SMOTE procedure generates synthetic data points for the minority (negative) category, whereas Tomek Links enhances class separation by discarding instances that are redundant or lie ambiguously between classes. After balancing, the dataset became evenly distributed, with 1,834 positive reviews and 1,834 negative reviews. With this balanced proportion, the model is expected to learn patterns from both classes more effectively and fairly.

TABLE IV
DATA DISTRIBUTION BEFORE AND AFTER BALANCING

Sentiment	Before-balancing	After-balancing
Positive	2.295	1.834
Negative	556	1.834
Total	**2.851**	**3.668**

F. Term Weighting

Word weighting using TF-IDF unigram was applied to identify the importance of terms in the user reviews. The results, presented in Figure 2, show the 20 terms with the highest weights, with "*mantap"* (0.1157), "*bagus"* (0.0690), and "*ok"* (0.0654) as the most dominant. These findings indicate that the majority of reviews express positive sentiment. In addition, terms such as *order*, *help*, and *application* highlight the functional aspects of the service, whereas words like *please* and *system* reflect criticism or user expectations for improvement.

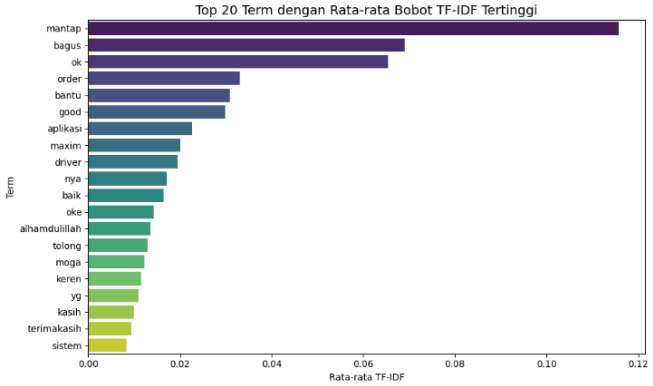

Figure 2. Average TF-IDF Values on the Maxim Application

The wordcloud visualization in Figure 3 illustrates the most frequently occurring words in user reviews. The terms *"driver," "order," "application,"* and *"Maxim"* appear most prominently, indicating users' focus on service-related aspects and application usage. Words such as *"good"* and *"great"* represent positive sentiment, whereas *"fictitious"* and *"difficult"* reflect user complaints about the system.

Figure 3. Wordcloud Visualization

G. Model Implementation and Evaluation

This study implements several classification algorithms, namely Support Vector Machine, K-Nearest Neighbor, and Random Forest, along with ensemble learning approaches through the Voting Classifier and the Combined Classifier. The implementation was conducted on a preprocessed dataset using 5-Fold Cross-Validation to ensure objective evaluation and to minimize bias arising from data partitioning.

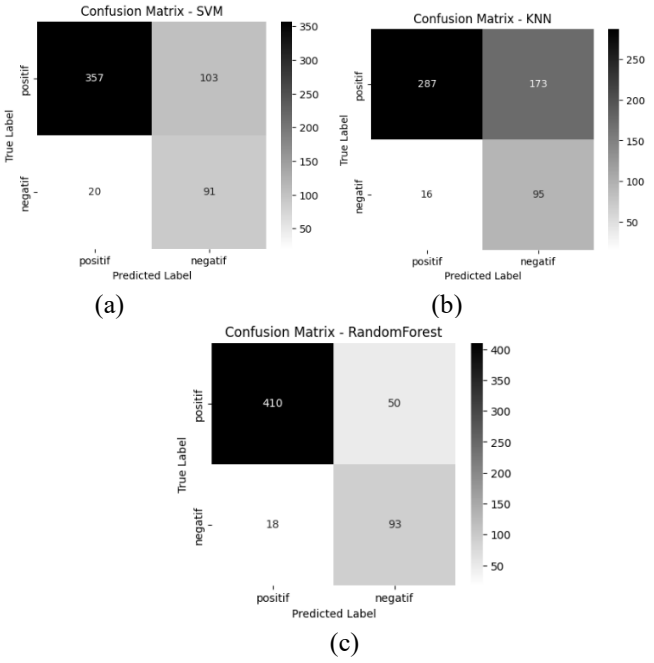

Figure 4. Confusion Matrix (a) SVM (b) K-NN (c) Random Forest

Based on the confusion matrix results, the Support Vector Machine (SVM) model (Figure 4a) successfully classified the majority of positive cases but showed a comparatively high error rate for negative cases. The K-Nearest Neighbor (KNN) model (Figure 4b) produced a higher number of errors in the positive class, resulting in lower accuracy compared to SVM. Meanwhile, the Random Forest (RF) model (Figure 4c) achieved the best performance among the individual models, with a more balanced prediction distribution between the positive and negative classes.

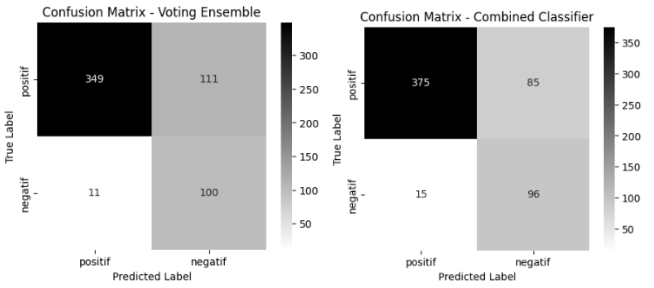

Figure 5. Confusion Matrix (a) Voting (b) Combined

For the ensemble approach, the Voting Classifier (Figure 5a) demonstrated moderate performance with relatively balanced errors across both classes, indicating stable classification ability. In contrast, the Combined Classifier (Figure 5b) showed a marked improvement over the Voting Classifier and nearly matched the performance of Random Forest.

Overall, the Combined Classifier produced more balanced results than the other methods, confirming that ensemble techniques—particularly combined classifiers can provide more stable performance compared to individual models.

TABLE V
CLASSIFICATION TEST RESULT WITH 5-FOLD

	Accuracy	Precision	Recall	F1
SVM	0.8004 ± 0.0331	0.8584 ± 0.0118	0.8004 ± 0.0331	0.8163 ± 0.0279
KNN	0.6945 ± 0.0294	0.8244 ± 0.0077	0.6945 ± 0.0294	0.7255 ± 0.0255
RF	0.8579 ± 0.0082	0.8787 ± 0.0085	0.8579 ± 0.0082	0.8647 ± 0.0079
Voting	0.8008 ± 0.0371	0.8645 ± 0.0149	0.8008 ± 0.0371	0.8174 ± 0.0318
Combined	0.8579 ± 0.0082	0.8787 ± 0.0085	0.8579 ± 0.0082	0.8647 ± 0.0079

The results of testing with 5-Fold Cross Validation, as presented in Table V, support this analysis. The Random Forest (RF) and Combined Classifier achieved the highest accuracy of 0.8579 ± 0.0082, along with precision of 0.8787 ± 0.0085, recall of 0.8579 ± 0.0082, and F1-score of 0.8647 ± 0.0079, indicating consistent and superior performance across all metrics. The SVM model obtained an accuracy of 0.8004 ± 0.0331, precision of 0.8584 ± 0.0118, recall of 0.8004 ± 0.0331, and F1-score of 0.8163 ± 0.0279, performing relatively better than KNN and the Voting Classifier, particularly in terms of precision. Meanwhile, the KNN model recorded the lowest accuracy at 0.6945 ± 0.0294 with an F1-score of 0.7255 ± 0.0255, reflecting its weakness in distinguishing classes effectively.

The Voting Ensemble model achieved an accuracy of 0.8008 ± 0.0371, precision of 0.8645 ± 0.0149, recall of 0.8008 ± 0.0371, and F1-score of 0.8174 ± 0.0318. Overall, its performance was slightly better than SVM but still did not surpass Random Forest (RF) and the Combined Classifier.

The experimental results indicate that the Random Forest (RF) and Combined Classifier algorithms achieved the best performance, with an accuracy of 85%, precision of 87%, recall of 85%, and an F1-score of 86%. The strong performance of RF can be attributed to its inherent ensemble nature, which effectively reduces overfitting and handles high-dimensional data efficiently. These findings are consistent with the results reported in [9] which identified RF as the most accurate algorithm for clinical prediction, and further support the conclusions in [4] that ensemble models demonstrate greater stability than single models.

The Combined Classifier approach, which integrates the prediction probabilities of SVM, KNN, and RF, has been shown to enhance both the accuracy and consistency of the results. This effectiveness arises from leveraging the unique strengths of each model: SVM excels in optimal margin separation, KNN captures local pattern structures, and RF efficiently handles feature variability. The combination mechanism balances the individual weaknesses of these models, resulting in more stable final outcomes, particularly on datasets with a high degree of initial imbalance. These findings are consistent with the study in [21], which demonstrated that combining voting and probabilistic methods yields classifications that are more robust to data variation.

Meanwhile, the SVM and Voting Classifier achieved moderate performance, with an accuracy of 80% and a precision of 86%. SVM tends to perform well on high-dimensional text data but remains sensitive to class imbalance. In contrast, KNN recorded the lowest performance, with an accuracy of 69%, due to its sensitivity to noise and the high dimensionality of TF-IDF features. These results are consistent with the findings in [6], which reported that KNN is less efficient for text data with a broad term distribution.

Overall, the results of this study reinforce that the ensemble learning approach particularly the Combined Classifier can enhance both the stability and accuracy of text classification compared to single models.

To further support these findings, the performance of each model was visualized through graphs comparing the average accuracy and ROC-AUC curves. This visualization clearly illustrates the performance differences among the algorithms in terms of both stability and their ability to distinguish between positive and negative classes. Figure 6 presents a comparison of the average accuracy of each model along with the standard deviation obtained from the 5-Fold Cross Validation results.

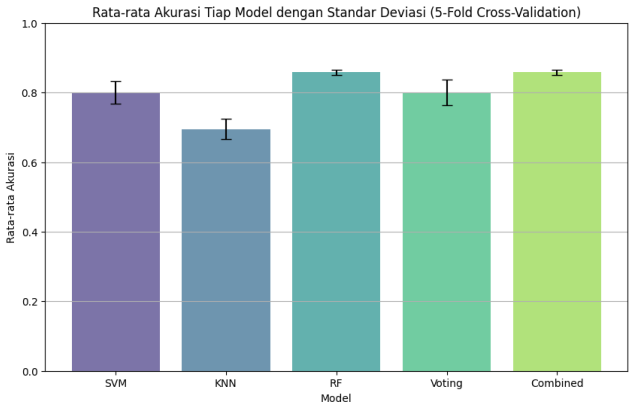

Figure 6. Graph of Maxim Review Data Analysis Results

Figure 6 presents a comparison of the average accuracy of each model along with the standard deviation obtained using 5-Fold Cross Validation. The KNN model recorded the lowest accuracy, while the Random Forest and Combined Classifier achieved the highest accuracy with relatively small deviations. Meanwhile, Figure 7 presents the ROC-AUC curve, illustrating each algorithm's ability to correctly classify data and distinguish between positive and negative sentiment classes.

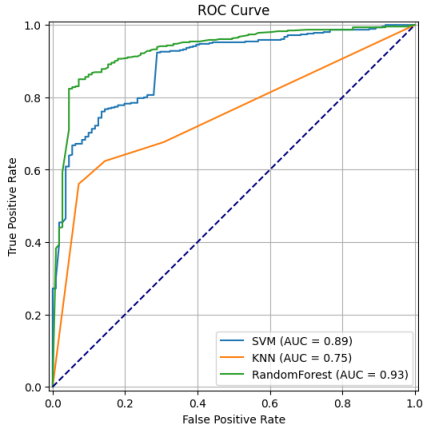

Figure 7. ROC-AUC Graph of Sentiment Analysis of Maxim Review Data

The corresponding AUC values are SVM = 0.89, KNN = 0.75, and Random Forest = 0.93. These results indicate that Random Forest demonstrates the highest capability in distinguishing between positive and negative classes, followed by SVM, while KNN exhibits relatively weaker separation performance. An AUC value above 0.8 suggests that all models generally possess good classification quality, although differences remain in terms of sensitivity and specificity across models.

Overall, the experimental results demonstrate that the Random Forest and Combined Classifier models deliver the most consistent performance across all evaluation metrics, whereas KNN exhibits the lowest performance due to its sensitivity to high-dimensional features and data imbalance. These findings further confirm that ensemble methods—particularly the Combined Classifier—enhance both the stability and accuracy of classification outcomes compared to single-model approaches [2] [4].

To ensure that the observed performance improvement was not due to random variation, a Wilcoxon signed-rank test was conducted on the F1-score values obtained from each fold of the 5-Fold Cross Validation for both models. The test resulted in a p-value of 0.125, indicating that there was no statistically significant difference between the performance of the Random Forest and Combined Classifier models at the 95% confidence level. Nonetheless, the Combined Classifier consistently achieved higher average evaluation metrics across all folds, suggesting that this ensemble approach offers greater predictive stability compared to the single model.

Consequently, the ensemble learning approach demonstrates strong potential for implementation in automated opinion analysis systems, enabling developers of online transportation applications to better understand user perceptions with greater accuracy and efficiency.

IV. CONCLUSION

This work investigated sentiment classification of Maxim app reviews on the Play Store by implementing Support Vector Machine, K-Nearest Neighbor, Random Forest, and ensemble learning methods through both the Voting and Combined Classifiers. The 5-Fold Cross Validation assessment indicated that Random Forest and the Combined Classifier attained superior outcomes, exhibiting higher accuracy, precision, recall, and F1-score in addition to enhanced stability over single-model approaches. Conversely, SVM and the Voting Classifier delivered intermediate results with notable strength in precision, whereas KNN demonstrated the lowest performance among the tested algorithms.

The results of this study confirm that ensemble learning, with the Combined Classifier in particular, significantly enhances sentiment analysis by providing higher accuracy and stability compared with standalone techniques. Furthermore, these outcomes affirm the value of ensemble methods in mitigating challenges arising from language heterogeneity, stylistic variations, and class imbalance in user-generated reviews of online transportation services.

Thus, future research can be extended by exploring other ensemble algorithms, such as Bagging, Boosting, and deep learning–based ensemble approaches to further enhance the generalization ability of the model. In addition, analyzing review data from multiple platforms and applying alternative data balancing techniques may provide more comprehensive and robust results.

REFERENCES

[1] U. Herni, "Analisis Sentimen dari Aplikasi Shopee Indonesia Menggunakan," *Indones. J. Appl. Stat.*, vol. 5, no. 1, pp. 31–38, 2022.

[2] M. J. Sai, P. Chettri, R. Panigrahi, A. Garg, A. K. Bhoi, and P. Barsocchi, "An Ensemble of Light Gradient Boosting Machine and Adaptive Boosting for Prediction of Type-2 Diabetes," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, 2023.

[3] Y. B. Lasotte, E. J. Garba, Y. M. Malgwi, and M. A. Buhari, "An Ensemble Machine Learning Approach for Fake News Detection and Classification Using a Soft Voting Classifier," *Eur. J. Electr. Eng. Comput. Sci.*, vol. 6, no. 2, pp. 1–7, 2022.

[4] K. Suresh Kumar *et al.*, "Sentiment Analysis of Short Texts Using SVMs and VSMs-Based Multiclass Semantic Classification," *Appl. Artif. Intell.*, vol. 38, no. 1, 2024.

[5] Idris I, Mustofa Y, and Salihi I, "Analisis Sentimen Terhadap Penggunaan Aplikasi Shopee Mengunakan Algoritma Support Vector Machine (SVM)," *Jambura J. Electr. Electron. Eng.*, vol. 5, pp. 32–35, 2023.

[6] F. Kurniawan and T. Supriyatno, "A contest of sentiment analysis: k-nearest neighbor versus neural network," *IAES Int. J. Artif. Intell.*, vol. 14, no. 2, pp. 1625–1633, 2025.

[7] K. Nugroho, E. Winarno, D. R. I. M. Setiadi, and O. Farooq, "Enhanced multi-lingual Twitter sentiment analysis using hyperparameter tuning k-nearest neighbors," *Bull. Electr. Eng. Informatics*, vol. 13, no. 6, pp. 4327–4334, 2024.

[8] J. J. Sanchez-Medina, "Sentiment analysis and random forest to classify LLM versus human source applied to Scientific Texts," pp. 1–12, 2024.

[9] J. Song *et al.*, "The random forest model has the best accuracy among the four pressure ulcer prediction models using machine

learning algorithms," *Risk Manag. Healthc. Policy*, vol. 14, pp. 1175–1187, 2021.

[10] Muhammad Nur Akbar, Nur Hasanahlmar'iyah Rusydi, M. Hasrul H., Nurul Shaumi Ramadhanti, and Erfiana, "Sentiment Analysis of Review Aplikasi Maxim di Google Play Store Menggunakan Support Vector Machine (SVM)," *AGENTS (Jurnal Sist. Informasi)*, vol. 2, no. 2, pp. 1–8, 2022.

[11] S. Syahrudin, Fenilinas Adi Artanto, Ahmad Rifqi Maulana, and F. Filsafat, "Metode Support Vector Machine (SVM) dan Lexicon-Based dalam Analisis Sentiment Ulasan Pengguna Aplikasi Wink," *JUMINTAL J. Manaj. Inform. dan Bisnis Digit.*, vol. 4, no. 1, pp. 59–73, 2025.

[12] F. T. Kurniati, D. H. Manongga, E. Sediyono, S. Y. J. Prasetyo, and R. R. Huizen, "Object Classification Model Using Ensemble Learning with Gray-Level Co-Occurrence Matrix and Histogram Extraction," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 9, no. 3, pp. 793–801, 2023.

[13] M. Periasamy, R. Mahadevan, B. L. S, R. C. Raman, H. K. S, and J. Jessiman, "Finding fake reviews in e-commerce platforms by using hybrid algorithms," 2024.

[14] N. V. R. Jhosefhin and C. Dewi, "Analisis Sentimen Crawling Data dari Sosial Media X tentang Gaza Menggunakan Metode SVM dan Decision Tree," *J. Indones. Manaj. Inform. dan Komun.*, vol. 6, no. 1, pp. 427–437, 2025.

[15] T. A. Zuraiyah, M. M. Mulyati, and G. H. F. Harahap, "Perbandingan Metode Naïve Bayes, Support Vector Machine Dan Recurrent Neural Network Pada Analisis Sentimen Ulasan Produk E-Commerce," *Multitek Indones.*, vol. 17, no. 1, pp. 27–43, 2023.

[16] F. Suandi *et al.*, *Enhancing Sentiment Analysis Performance Using SMOTE and Majority Voting in Machine Learning Algorithms*, no. Icae 2024. Atlantis Press International BV, 2024.

[17] Y. A. Mustofa, I. Surya, and K. Idris, "Pendekatan Ensemble pada Analisis Sentimen Ulasan Aplikasi Google Play Store Ensemble Approach to Sentiment Analysis of Google Play Store App Reviews," *Jambura J. Electr. Electron. Eng.*, vol. 6, no. 2, pp. 181–188, 2024.

[18] T. N. Wijaya, R. Indriati, and M. N. Muzaki, "Analisis Sentimen Opini Publik Tentang Undang-Undang Cipta Kerja Pada Twitter," *Jambura J. Electr. Electron. Eng.*, vol. 3, no. 2, pp. 78–83, 2021.

[19] R. Sakti *et al.*, "Review of Literature on Improving the KNN Algorithm," *Trans. Mach. Learn. Artif. Intell.*, vol. 11, no. 3, pp. 63–72, 2023.

[20] A. Alsayat, "Improving Sentiment Analysis for Social Media Applications Using an Ensemble Deep Learning Language Model," *Arab. J. Sci. Eng.*, vol. 47, no. 2, pp. 2499–2511, 2022.

[21] D. Ghoul, J. Patrix, G. Lejeune, and J. Verny, "A combined AraBERT and Voting Ensemble classifier model for Arabic sentiment analysis," *Nat. Lang. Process. J.*, vol. 8, no. December 2023, p. 100100, 2024.

[22] L. Rohmatun and A. Baita, "Machine Learning-Based Sentiment Analysis on Twitter (X): A Case Study of the ' Kabur Aja Dulu ' Issue Using SVM," vol. 9, no. 4, pp. 1972–1983, 2025.