

Sentiment Analysis of Tokopedia App Reviews using Machine Learning and Word Embeddings

Muhammad Idris¹⁾, Ahmad Rifai^{2)*}, Ken Ditha Tania³⁾

¹⁾²⁾³⁾Faculty of Computer Science, Universitas Sriwijaya, Indonesia

¹⁾idrist2003@gmail.com, ²⁾rifai.bae@gmail.com, ³⁾kenya.tania@gmail.com

Submitted :29 Nov 2024 | **Accepted** : Dec 20, 2024 | **Published** : Jan 8, 2025

Abstract: Tokopedia, a prominent e-commerce platform in Indonesia, generates vast amounts of user feedback, offering valuable insights into customer satisfaction through sentiment analysis. However, sentiment analysis of app reviews specifically on Tokopedia reviews remains underexplored. Sentiment analysis, also known as opinion mining, categorizes user sentiments into positive or negative, offering insights into user preferences and service shortcomings. While traditional text representation techniques like TF-IDF are widely used for sentiment analysis, they lack the semantic richness provided by advanced word embeddings such as Word2Vec and FastText, which excel at capturing contextual relationships between words. These methods have shown potential to enhance the performance of machine learning models in sentiment analysis tasks. This study evaluates the performance of three machine learning algorithms—Support Vector Machine (SVM), Random Forest (RF), and Gaussian Naïve Bayes (NB)—combined with Word2Vec and FastText feature extraction. A dataset of 59,811 Tokopedia app reviews was scraped from the Google Play Store, preprocessed, and subjected to feature extraction using Word2Vec and FastText. SVM achieved the best performance, with an accuracy of 89.06% using FastText and 89.02% using Word2Vec. RF ranked second with accuracies of 88.07% for FastText and 88.14% for Word2Vec. NB showed the lowest performance, achieving 84.26% with Word2Vec and 83.73% with FastText. Differences in performance between Word2Vec and FastText embeddings were minimal across all algorithms, highlighting their comparable effectiveness. These results underscore SVM's consistent superiority across various configurations for sentiment analysis.

Keywords: FastText, Word2Vec, Naïve Bayes, Random Forest, Support Vector Machine, Sentiment Analysis, Tokopedia

INTRODUCTION

Tokopedia is an e-commerce platform that has spearheaded Indonesian retail's e-commerce evolution. It made online shopping more straightforward for both consumers and sellers. As of March 2021, Tokopedia's market share was 32.04%, followed by Shopee at 29.73% and the other in subsidiaries (Susilawati, 2021). The platform gained increased prominence as shoppers started looking for alternate means of online shopping that are safer and easier to use (Pangilinan et al., 2023). By leveraging the services of payment gateways and working with logistics providers, Tokopedia can assist its customers, resulting in high satisfaction on the platform (Pangilinan et al., 2023).

However, given that there are millions of users and thousands of comments, it is important to understand how people think to determine how satisfied users are with the product or service and where we can improve by performing user sentiment analysis on the comments left on the platform. Therefore, using sentiment analysis allows developers to categorize the feedback provided by users, look for positive and negative experiences to assess engagement, and adjust the services accordingly to the users' expectations (Garousi & Cutting, 2021).

Sentiment analysis, also known as opinion mining, is the practice of analyzing opinions, sentiments, or emotions about a particular document or text concerning any product, service, or device (Liu, 2022). Analyzing sentiment in user reviews is helpful since it gives an idea of users' expectations and experiences, which could

*name of corresponding author



change rapidly (Karunia et al., 2024). This understanding is important to ensure that services in the ever-evolving world of e-commerce are customized to meet users' expectations.

Numerous studies have utilized machine learning algorithms such as Support Vector Machines (SVM), Naïve Bayes (NB), and Random Forests (RF) for sentiment analysis, demonstrating their effectiveness across various datasets and tasks. Traditional text representation methods like Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) are commonly used in sentiment analysis. However, BoW disregards word context and creates large, sparse matrices, leading to computational inefficiencies, while TF-IDF focuses on word importance but neglects semantic relationships (Royyan & Setiawan, 2022; Khomsah et al., 2022).

Word embeddings address these limitations by representing words based on their semantic relationships. Word2Vec, introduced by Mikolov et al. in 2013, maps words to dense vector spaces that capture contextual relevance, while FastText, developed by Facebook AI Research, improves on Word2Vec by incorporating subword information, benefiting languages with complex morphology (Khomsah et al., 2022). These advancements enhance sentiment analysis by capturing deeper contextual meanings.

The effectiveness of word embeddings is also proved by the increase of accuracy compared to the baseline method from previous researches, such as (Damayanti & Lhaksana, 2024) showed better results for SVM and Word2Vec compared to TF-IDF for sentiment analysis on Twitter's 2024 Indonesia President election with the accuracy reaching 90%, and (Raihan & Setiawan, 2022) also showed the significant improvement after the SVM and FastText feature expansion applied compared to SVM-TF-IDF as the baseline method. Another research by showed promising results for Word2vec and FastText feature extraction with RF and Extra Tree algorithm. Research from (Islamanda & Sibaroni, 2024) evaluated the performance of Word2Vec Skip Gram and CBOW architecture with four different NB algorithms, emphasizing that the best results were achieved with Gaussian NB and Skip Gram model with 77.18% accuracy.

This study aims to perform sentiment analysis on Tokopedia app reviews on the Google Play Store platform, a specific focus area that has received limited attention based on a preliminary search of existing literature. It focuses on evaluating the performance of three machine learning algorithms: SVM, Gaussian NB, and RF. This study focused on utilizing advanced word embeddings, such as Word2Vec and FastText, as feature extraction methods, which have been proven to enhance classification accuracy in previous sentiment analysis tasks. The results will be obtained using the classification report metrics, such as accuracy, precision, recall, and F1 score. The visualization of Word Cloud will be added to visualize the occurrences of the words in the reviews dataset. By comparing the performance of these algorithms and embeddings, this research contributes to the field by identifying the most effective combination for sentiment classification of e-commerce app reviews.

LITERATURE REVIEW

Algorithms based on Support Vector Machine (SVM), Random Forest (RF), and Naive Bayes (NB) have received significant attention from researchers, especially in terms of text analysis and classification problems. Research by (Chong & Shah, 2022) examined the performance of NB and SVM classifiers for sentiment analysis on healthcare stock comments the stock. After using the Grid Search technique to tune hyperparameters, SVM was better than NB, resulting in 85.65% and 68.70% accuracy, respectively. The results also suggested that SVM had better optimization ability as it only improved by 4% accuracy, while NB improved by only 1%. Similarly, (Kristiyanti & Sri Hardani, 2023) confirmed the applicability of SVM and NB algorithm with TF-IDF for analyzing public sentiment, and the conclusion is that SVM gave the best result, having an accuracy rate of 84.89% and F1 score of 92%. For the NB, the performance of similar metrics compared to SVM was competitive, indicating its usefulness for these tasks.

RF has also been applied successfully in the area of sentiment analysis. A study from (Asian et al., 2022), RF and Multi-Layer perceptron (MLP) were utilized to analyze the sentiment of a Brazilian anesthesiologist with 96.42 % accuracy when RF with TF-IDF features was applied. Similarly, (Sumertajaya et al., 2022) applied RF and SVM to analyze sentiments towards COVID-19 vaccines. Although SVM performed better than RF with an F1-score of 0.52 while RF scored 0.51, both algorithms are said to be able to deal with sophisticated sentiment datasets.

Word embeddings like Word2Vec and FastText have improved the quality of feature representation thus improving sentiment analysis. For instance, (Damayanti & Lhaksana, 2024) implemented Word2Vec in SVM for the sentiment analysis of the Indonesian presidential candidates' elections, whereby they achieved the highest accuracy rate of 90.75% from Word2Vec method compared to TF-IDF method with highest accuracy at only 82.3%. Research by (Rezki et al., 2023) similarly reported the high accuracy of 89.87% by employing Word2Vec with SVM for sentiments towards the "Merdeka Belajar Kampus Merdeka" policy. Using NB, (Islamanda & Sibaroni, 2024) have conducted sentiment analysis on Indonesia's Whoosh train service using Word2Vec and various NB variations. They reported good results, achieving 77.18% accuracy for Gaussian NB. Such studies prove that the application of Word2Vec can boost the performance of various sentiment analysis tasks.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

FastText, known for handling subword information, has also gained traction. Research by (Raihan & Setiawan, 2022) utilized FastText embeddings with SVM for aspect-based sentiment analysis of Telkomsel’s products, achieving impressive F1-scores of over 94% across different aspects. The ability of FastText to address out-of-vocabulary words has proven advantageous in tasks involving informal language or typos.

Some comparative studies examine the strengths and weaknesses of models and word embeddings. One of them is from (Royyan & Setiawan, 2022), analyzed public policy sentiment using SVM and Logistic Regression (LR) along with Word2Vec embeddings for feature expansion, where SVM was the best model in the study, attaining an accuracy level of 78.99%. In another research, (Khomsah et al., 2022) test the performance of Word2Vec and FastText models with four different kinds of tree classifier models, including Decision Tree (DT), RF, Extra Tree (ET), and AdaBoost in the task of sentiment analysis, using a dataset containing 3145 Trip Advisor hotel reviews, and report that Random Forest and Extra Tree classifiers which employed FastText performed best with accuracy levels above 93% compared to 85% achieved by the baseline Decision Tree model.

METHOD

In this research, a sentiment classification system will be developed to analyze Tokopedia app user opinions expressed in text. The dataset, sourced from an e-commerce platform, will be used to compare the accuracy of various text feature extraction techniques combined with the SVM, RF, and NB algorithm. Word2Vec and FastText will be evaluated to identify the most effective approach for sentiment analysis. Figure 1 illustrates the overall system design and methodology (Khomsah et al., 2022).

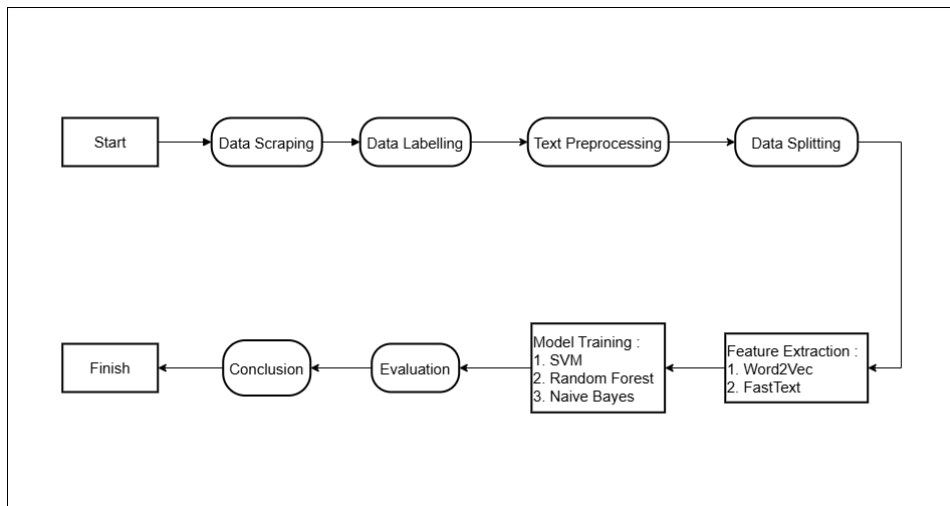


Fig. 1 Research Methodology Flowchart

Data Scraping

In this first step, we scrape Tokopedia app review data from the Google Play Store with the help of google-play-scraper, a Python package to enhance the scraping process to extract Indonesian and any other language reviews, including ratings, text reviews, data created, and metadata. The data will be saved in an ordered CSV file containing fields for review text, rating, thumbs-up count, and review date, but only the review text and ratings were kept for further analysis. Once we collected the data, we performed the data cleaning steps, including eliminating null values, duplicates, and rating outliers, reducing the data count to 59,811.

Table 1. Data scraping result

Scrapped data	Data after cleaning
60,000	59,811

Data Labeling

Data labeling is conducted to point out the texts' positive, negative, and/or neutral emotional values (Regina et al., 2023). In this case, we will be looking at the data from the perspective of this study using review score data.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Scores between one and three will be labeled negative (0), while all scores above three will be labeled positive (1). This step will be done automatically by implementing Pandas 'apply' function to determine and add the label class. Some examples of labeling can be seen in Table 2.

Table 2. Review Labeling Result Example

Review Text	Score	Sentiment
“Saya sebenarnya dulu senang belanja di tokopedia saat masih bisa pilih ekspedisi yang kita inginkan walaupun gratis ongkir. Tp sekarang sangat mengecewakan karena kita tidak lagi bisa pilih ekspedisi yang kita inginkan dan selalu dikasih ekspedisi yang super lelet. Ini terakhir saya belanja di tokopedia kalau memang masih tidak bisa kita pilih ekspedisi.”	1	Negative
“Setiap kali mau pakai voucher potongan promo, eh malah dibatalkan oleh sistem tokopedia. Pakai 2 akun yang berbeda pun juga sama saja. Mana vouchernya tidak bisa kembali , hmm mengecewakan merasa dirugikan.”	2	Negative
“Sebenarnya pake tokopedia sudah lama. Tapi akhir2 ini, sejak ada kurir rekomendasi saya jadi males. Karena pengiriman jadi lama. Bisa pindah2 ekspedisi gitu. Kalopun ga pindah2 juga ekspedisi yang dipake lemotnya minta ampun, kayak ID express dan SAPX. Kurir rekomendasinya ga kompeten banget”	3	Negative
“Aplikasinya bagus tp syg utk pengiriman jne sering mslh trs..ud beberapa kali pakai jne info brg sampai tp tdk saya terima..perbaiki utk pengiriman nya..diinfokan ke pihak jne nya”	4	Positive
“mantab promonya banyak. jangan sampai hilang ya promonya. karena sangat membantu. makasih”	5	Positive

Text Preprocessing

Preprocessing techniques greatly influence sentiment analysis, as they are closely tied to the outcomes generated (Hidayat et al., 2022). Preprocessing aims to eliminate text noise and spelling errors while enhancing the data's machine readability to minimize ambiguity during feature extraction (Raihan & Setiawan, 2022). In this study, the data preprocessing procedures encompass a series of actions. In the first place, case folding is used to change all texts into lowercase. Then, regular expressions (regex) help to remove punctuation, superscripts, uncommon characters, correct excessive alphabetical repetition at the first and last position of the text, and remove extra spaces. Slang words are standardized by converting nonstandard or informal terms into proper forms. Afterward, tokenization is done to cut the text down into tokens. The tokenized texts then went through the process of stop words removal to reduce the unimportant vocabularies that might add more noises to the feature extraction process. Several Python libraries, such as re and NLTK, support these processes. The example of preprocessing results can be seen in Table 2, and the data distribution can be seen in Table 3.

Data Splitting

Splitting data into smaller but meaningful components that can easily be worked on and allowing machine learning models to handle text data effectively is one of the most basic tasks of NLP (Das et al., 2023). In this case, the researcher employs an 80:20 split whereby eighty percent is used to build the model while twenty percent is left to assess or test the model's performance developed with part of the data. The result of data splitting can be seen in Table 4.

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Table 3. Text Preprocessing Result Example

Preprocessing Step	Text After Preprocessing
Original text	“Krn dana sdh dikembalikan, rating sy naikkan lg. Cm msh krg suka sm tampilan tokoed skrg, lbh bagus yg dulu, byk brg yg g dicari suka muncul disearch, minta brg diurutin harga tp ttp aja ada yg ngacak-ngacak”
Case Folding	“krn dana sdh dikembalikan, rating sy naikkan lg. cm msh krg suka sm tampilan tokoed skrg, lbh bagus yg dulu, byk brg yg g dicari suka muncul disearch, minta brg diurutin harga tp ttp aja ada yg ngacak-ngacak”
Cleaning using RegEx	“krn dana sdh dikembalikan rating sy naikkan lg cm msh krg suka sm tampilan tokped skrg lbh bagus yg dulu byk brg yg g dicari suka muncul disearch minta brg diurutin harga tp ttp aja ada yg ngacak ngacak”
Slang words normalization	“karena dana sudah dikembalikan rating saya naikkan lagi cuma masih kurang suka sama tampilan tokopedia sekarang lebih bagus yang dulu banyak barang yang tidak dicari suka muncul di search minta barang diurutkan harga tapi tetap saja ada yang mengacak mengacak”
Tokenization	['karena', 'dana', 'sudah', 'dikembalikan', 'rating', 'saya', 'naikkan', 'lagi', 'cuma', 'masih', 'kurang', 'suka', 'sama', 'tampilan', 'tokopedia', 'sekarang', 'lebih', 'bagus', 'yang', 'dulu', 'banyak', 'barang', 'yang', 'tidak', 'dicari', 'suka', 'muncul', 'di', 'search', 'minta', 'barang', 'diurutkan', 'harga', 'tapi', 'tetap', 'saja', 'ada', 'yang', 'mengacak', 'mengacak']
Stopwords removal	['karena', 'dana', 'sudah', 'dikembalikan', 'rating', 'naikkan', 'lagi', 'cuma', 'kurang', 'suka', 'tampilan', 'tokopedia', 'bagus', 'dulu', 'banyak', 'barang', 'tidak', 'dicari', 'suka', 'muncul', 'search', 'minta', 'barang', 'diurutkan', 'harga', 'tetap', 'saja', 'mengacak', 'mengacak']

Table 4. Data splitting distribution result

Subset	Total data	Positive	Negative
Training	47,848	22,418	25,430
Test	11,963	5,658	6,305

Feature Extraction

The emphasis in the feature extraction step relies on applying different text representation techniques that allow the transformation of the unstructured text data into a structured format or vectors suitable for the chosen ML model. In this case, this research study considers two methods: Word2Vec and FastText. In addition, default parameters are also set to balance the quality of feature representation for the embedding methods: using the skip-gram model, a window size of 5 is used to define the context range, a min_count of 5 filters out infrequent words, and a dimension of 100 specifies the length of the feature vectors.

Word2Vec

Word2Vec is a method of embedding words to vectors introduced by Mikolov et al. in 2013, which allows to transform words into numerical patterns that describe the meaning of the words as well as their relationships to each other (Damayanti & Lhaksana, 2024). There are two different models of Word2Vec: the Continuous Bag-of-Words (CBOW) model and the Skip-Gram model (Royyan & Setiawan, 2022). The CBOW model predicts target words using information from their surrounding context, thus making it generally quicker and more efficient when handling high-frequency words. In contrast, the Skip-Gram model uses a single predicted word in the

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

prediction of context, thus being effective while working with fewer data resources for training and, in addition, being able to represent less frequent words (Damayanti & Lhaksana, 2024). For this study, we will employ the Word2Vec module from the Gensim Python Library to train and generate the Word2Vec word vectors.

FastText

FastText is a word embedding algorithm introduced by Facebook, which encodes words in terms of their constituent subwords or n-grams (Raihan & Setiawan, 2022). In contrast to Word2Vec, which considers each word as a single component, FastText incorporates word subcomponents in the form of character n-grams to embed the structure of words, such as their stems or affixes. This feature also makes FastText more suitable for complex morphological languages, enabling the model to address issues such as Out-Of-Vocabulary (OOV), sparse, and misspelled words more efficiently (Khomsah et al., 2022). Nevertheless, both models relate to the same overall target, word representation, only that FastText leads to a better representation but at the cost of larger vector dimensions and processing time (Khomsah et al., 2022). This research will use the fasttext Python library to fit and produce the FastText word vectors.

Support Vector Machine (SVM)

SVM can be applied to varied classification models, such as sentiment classification. In the development of SVM, Boser, Guyon, and Vapnik were the pioneers when they presented SVM in 1992 where it has been validated in the field and is known to solve problems better than most methods (Damayanti & Lhaksana, 2024). The SVM algorithm is designed to search for the best-dividing hyperplane between classification groups, which are, in this case, positive and negative groups, by maximizing the margin, which is the distance from the hyperplane to the closest data point known as support vectors (Royyan & Setiawan, 2022). In this experiment, the SVM will be used as one of the classifiers using the sklearn Python library with a linear kernel, and no other hyperparameters will be altered. We will train four SVM models, where each SVM model will be trained separately using feature vectors generated by the Word2Vec and FastText feature extraction techniques.

Gaussian Naïve Bayes

Gaussian NB is a variant of the NB algorithm that is particularly suited for continuous data. It assumes that the continuous values associated with each class are distributed according to a Gaussian (normal) distribution. This model is adequate for classification tasks where the features are continuous rather than categorical (Islamanda & Sibaroni, 2024). Given this condition, there are chances that this algorithm can efficiently use the presentation of continuous word vectors from Word2Vec and FastText. In this experiment, Gaussian NB will be used as one of the classifiers. The model will be applied using the sklearn Python library without altering its default parameters. Two separate Gaussian Naive Bayes models will be trained, each utilizing feature vectors generated by the Word2Vec and FastText feature extraction techniques.

Random Forest (RF)

RF is an ensemble approach of CART (Classification and Regression Trees), a kind of decision tree algorithm (Asian et al., 2022). RF creates multiple Decision Trees from different subsets of the training data (bootstrap samples) based on the ensemble learning principle (Khomsah et al., 2022). RF is an ensemble learning method that creates multiple decision trees by resampling the training data with replacement (bootstrapping) and randomly sampling features for each tree's splits (Asian et al., 2022). This methodology reduces the chances that the output will fit a training set but not the unseen data, also known as the overfitting problem (Khomsah et al., 2022). In this experiment, Random Forest will be employed as one of the classifiers. The model will be applied with the sklearn Python library using its default parameters without further tuning. Two separate RF models will be trained, each using feature vectors derived from the Word2Vec and FastText feature extraction techniques.

Evaluation

The classification report and confusion matrix visualization are the most common sentiment analysis evaluation methods. The classification report cites the accuracy, precision, recall, and F1 score, enabling us to assess the model's usefulness in these measures, and the confusion matrix illustrates all the classes and their actual and false predictions, which enables us to know frequent cases of wrong predictions. This research will use the classification report and confusion matrix as the evaluation method. We will also use the Word Cloud visualization to visualize the frequently occurring words and understand which words mostly appeared on positive and negative class reviews. These methods ultimately offer transparent decision-making concerning the completeness of the model and its weaknesses in sentiment analysis.

*name of corresponding author



This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

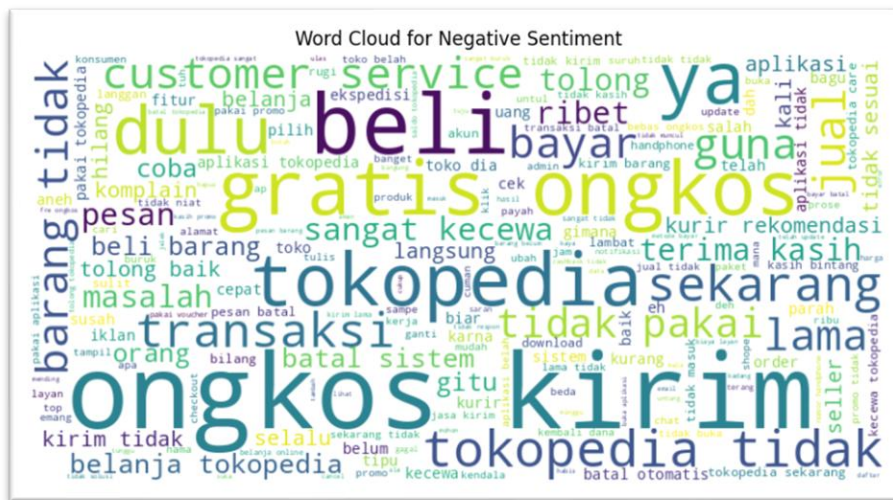


Fig. 3 Word Cloud for Negative Class

We can see from Figure 3, the word occurrences from negative class reviews containing various words, including “ongkos”, “kirim”, “gratis”, “tokopedia”, “tidak”, “beli”, and many more.

DISCUSSIONS

As we can see from Table 5 and Table 6, SVM’s performance was the best across all evaluation metrics using Word2Vec and FastText feature extraction. With FastText, SVM performed with an 89.06% accuracy, 89.81% precision, 88.66% recall, and 88.90% F1- score. When combined with Word2Vec feature extraction, SVM resulted in values of accuracy, precision, recall, and F1 score of 89.02%, 89.78%, 88.61%, and 88.90%, respectively. The results showed that when using SVM, there is no significant change in performance between Word2Vec and FastText as the embedding.

The second-best-performing model, RF, using Fast Text and Word2Vec, gave an accuracy of 88.07% and 88.14%, respectively. These results showed that embedding Word2Vec was slightly better than FastText for RF. In addition, for Gaussian NB, the accuracy of Word2Vec (84.26%) was slightly better than Fast Text (83.73%), as well for precision and recall values.

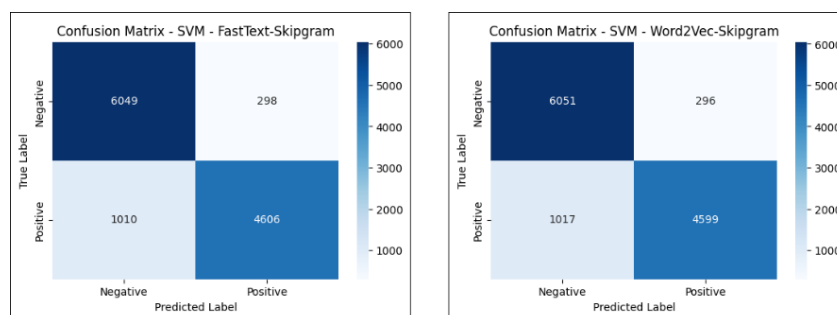


Fig. 4 SVM Confusion Matrix

The confusion matrix in Figure 2 provides a detailed breakdown of how well SVM, the best-performing model, performed with FastText and Word2Vec embeddings by categorizing the predictions into four key groups: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). Both feature extraction methods show strong performance, with very slim differences among the TP, TN, FP, and FN.

These results are similarly aligned with a study from (Khomsah et al., 2022) that resulted in slight and insignificant differences between the combination of various ML algorithms and word embeddings, which might have occurred because of the data imbalance of the researchers’ dataset. In the case of our dataset, these results might also occur because our dataset is limited to only Tokopedia reviews, which have some unique vocabularies that present both positive and negative sentiment, which could confuse the model to distinguish the word differences. For example, the words “tokopedia”, “ongkos”, and “kirim” in Figures 2 and 3 frequently appeared in

*name of corresponding author



both positive and negative classes, which showed the pros and cons regarding the free shipping service from Tokopedia users.

CONCLUSION

This research conducted sentiment analysis of Tokopedia app reviews using Support Vector Machine (SVM), Random Forest (RF), and Gaussian Naïve Bayes (NB) algorithm and word embeddings feature extraction methods (Word2Vec and FastText). The results showed that SVM yielded the best accuracy, followed by RF as second and Gaussian NB as the third best-performing model on both feature extraction methods. The best combination of ML algorithms and word embeddings is SVM-FastText, with the highest accuracy of 89.06%.

However, the accuracy and other metrics differences between Word2Vec and FastText feature extraction across all chosen ML algorithms are slight and insignificant. This could be due to the dataset imbalance and the occurrence of some vocabularies that appear frequently on both positive and negative reviews, which might confuse the model when determining the sentiment of those words.

Future research should evaluate the accuracy of different hyperparameters for ML methods using Grid Search or much more complex classification algorithms, such as Deep Learning. A more diverse vocabulary dataset, such as social media, and more robust word embeddings for feature extraction, such as IndoBERT, can be used to compare which method is better.

REFERENCES

- Asian, J., Rosita, M. D., & Mantoro, T. (2022). Sentiment Analysis for the Brazilian Anesthesiologist Using Multi-Layer Perceptron Classifier and Random Forest Methods. *Jurnal Online Informatika*, 7(1), 132–141. <https://doi.org/10.15575/join.v7i1.900>
- Chong, K., & Shah, N. (2022). Comparison of Naive Bayes and SVM Classification in Grid-Search Hyperparameter Tuned and Non-Hyperparameter Tuned Healthcare Stock Market Sentiment Analysis. *International Journal of Advanced Computer Science and Applications*, 13(12), 90–94. <https://doi.org/10.14569/IJACSA.2022.0131213>
- Damayanti, L., & Lhaksmaha, K. M. (2024). Sentiment Analysis of the 2024 Indonesia Presidential Election on Twitter. *Sinkron*, 8(2), 938–946. <https://doi.org/10.33395/sinkron.v8i2.13379>
- Das, R. K., Islam, M., Hasan, M. M., Razia, S., Hassan, M., & Khushbu, S. A. (2023). Sentiment analysis in multilingual context: Comparative analysis of machine learning and hybrid deep learning models. *Heliyon*, 9(9), e20281. <https://doi.org/10.1016/j.heliyon.2023.e20281>
- Garousi, V., & Cutting, D. (2021). What do users think of the UK's three COVID-19 contact-tracing apps? A comparative analysis. *BMJ Health & Care Informatics*, 28(1), e100320. <https://doi.org/10.1136/bmjhci-2021-100320>
- Hidayat, T. H. J., Ruldeviyani, Y., Aditama, A. R., Madya, G. R., Nugraha, A. W., & Adisaputra, M. W. (2022). Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier. *Procedia Computer Science*, 197, 660–667. <https://doi.org/10.1016/j.procs.2021.12.187>
- Islamanda, M. D., & Sibarani, Y. (2024). Whoosh User Sentiment Analysis on Social Media Using Word2Vec and the Best Naïve Bayes Probability Model. *Sinkron*, 8(3), 1558–1568. <https://doi.org/10.33395/sinkron.v8i3.13742>
- Karunia, K., Putri, A. E., Fachriani, M. D., & Rois, M. H. (2024). Evaluation of the Effectiveness of Neural Network Models for Analyzing Customer Review Sentiments on Marketplace. *Public Research Journal of Engineering, Data Technology and Computer Science*, 2(1), 52–59. <https://doi.org/10.57152/predatecs.v2i1.1100>
- Khomsah, S., Ramadhani, R. D., & Wijaya, S. (2022). The Accuracy Comparison Between Word2Vec and FastText On Sentiment Analysis of Hotel Reviews. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(3), 352–358. <https://doi.org/10.29207/resti.v6i3.3711>
- Kristiyanti, D. A. & Hardani, S. (2023). Sentiment Analysis of Public Acceptance of Covid-19 Vaccines Types in Indonesia using Naïve Bayes, Support Vector Machine, and Long Short-Term Memory (LSTM). *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 7(3), 722–732. <https://doi.org/10.29207/resti.v7i3.4737>
- Liu, B. (2022). *Sentiment Analysis and Opinion Mining*. Springer Nature.
- Pangilinan, G. A., Tambunan, A., & Astuti, E. D. (2023). Tokopedia E-Commerce is Being Used to Present Opportunities for Young Business Owners to Succeed in the Digital Economy Amid the Pandemic. *Startuppreneur Business Digital (SABDA Journal)*, 2(2), 182–191. <https://doi.org/10.33050/sabda.v2i2.284>

*name of corresponding author



This is anCreative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

- Raihan, M. A. & Setiawan, E. B. (2022). Aspect Based Sentiment Analysis with FastText Feature Expansion and Support Vector Machine Method on Twitter. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(4), 591–598. <https://doi.org/10.29207/resti.v6i4.4187>
- Regina, R., Saragih, T. H., & Kartini, D. (2023). ANALISIS SENTIMEN BRAND AMBASSADOR BTS TERHADAP TOKOPEDIA MENGGUNAKAN KLASIFIKASI BAYESIAN NETWORK DENGAN EKSTRAKSI FITUR TF-IDF. *Jurnal Informatika Polinema*, 9(4), 383–390. <https://doi.org/10.33795/jip.v9i4.1333>
- Royyan, A.R. & Setiawan, E.B. (2022). Feature Expansion Word2Vec for Sentiment Analysis of Public Policy in Twitter. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(1), 78–84. <https://doi.org/10.29207/resti.v6i1.3525>
- Rezki, N., Thamrin, S. A., & Siswanto, S. (2023). SENTIMENT ANALYSIS OF MERDEKA BELAJAR KAMPUS MERDEKA POLICY USING SUPPORT VECTOR MACHINE WITH WORD2VEC. *BAREKENG: Jurnal Ilmu Matematika Dan Terapan*, 17(1), 481–486. <https://doi.org/10.30598/barekengvol17iss1pp0481-0486>
- Sumertajaya, I. M., Angraini, Y., Harahap, J. R., & Fitrianto, A. (2022). Sentiment Analysis on Covid-19 Vaccination in Indonesia Using Support Vector Machine and Random Forest. *JUITA: Jurnal Informatika*, 10(1), 1. <https://doi.org/10.30595/juita.v10i1.12394>
- Susilawati, E. (2021). The Influence of Mobile Banking Easiness and Flash Sale towards Impulse Buying on Shopee Users in Bandung. *Proceeding of International Conference on Business, Economics, Social Sciences, and Humanities*, 4, 333–338. <https://doi.org/10.34010/icobest.v2i.291>