 ISSN 2579-7298

The Classification Method is Used for Sentiment Analysis in My Telkomsel

Deni Hardiansyah[a], RZ Abdul Aziz[a], Muhammad Said Hasibuan[a]

[a] Institut Informatika dan Bisnis Darmajaya, Jl. ZA Pagar Alam No. 93 A, Lampung 35136, Indonesia
rz_aziz@darmajaya.ac.id
* corresponding author

ARTICLE INFO

Article history
Received 03 June 2024
Revised 29 Aug 2024
Accepted 30 Sep 2024

Keywords
Sentiment Analysis,
Mobile Applications,
MyTelkomsel,
Natural Language Processing (NLP)

ABSTRACT

User reviews significantly impact how mobile apps are perceived and provide developers with valuable insights into improving the functionality and quality of their products. Sentiment analysis of these evaluations helps identify the main issues faced by consumers, such as technical difficulties, costs, and service levels. The main objective of this study is to classify user sentiment into positive and negative categories, focusing on the MyTelkomsel app. With the use of Google Play Scraper, 39,493 reviews on various app versions and user experiences were collected. This data was analyzed using multiple machine learning models, including Support Vector Machines (SVM), Naive Bayes, Random Forest, and Gradient Boosting, alongside the Natural Language Processing (NLP) approach. The results show that 39.2% of the reviews are positive, while 60.8% reflect negative sentiment. Among the models, SVM showed the highest accuracy in sentiment classification with a value of (0.854792), while Naive Bayes (0.775541), Random Forest (0.829725), and Gradient Boosting (0.819344) also performed well in sentiment classification. These findings suggest that developers can leverage the insights gained from this analysis to proactively improve the performance and user experience of the MyTelkomsel app, by addressing technical and service-related issues identified in user reviews.

1. Introduction

Industrials Revolution 4.0 is an opportunity for Indonesia to increase economic growth [1]. Technology is so important for human life because it can improve and make our lives easier [2]. The use of mobile applications has become ingrained in the daily lives of people around the world in this increasingly sophisticated digitalization era. The latest statistics show that there are 6.4 billion smartphone users globally in 2021, and the use of mobile applications for various functions, including communication, entertainment, and financial transactions, is increasing significantly every year[3]. More than 100 million people use mobile applications every day in Indonesia, where smartphone adoption has exceeded 70% of the population [4].

User reviews have a significant impact on how people view mobile applications and provide developers with valuable information on how to improve the functionality and quality of their products. Previous studies have shown the importance of using user feedback to identify issues, improve functionality, and enhance the overall user experience [3]. Sentiment analysis, a component of Natural Language Processing (NLP), recognizes sentiment patterns in review text to help extract deeper insights from user evaluations [5].

Mobile applications such as MyTelkomsel in Indonesia have particular difficulties because the country's diverse language population, use of slang, dialects, and regional expressions can distort sentiment analysis results. To accurately understand user sentiment and incorporate local subtleties, sentiment analysis must use a more flexible approach [6].

MyTelkomsel rating data on Google Playstore, Number of downloaders 100 million MyTelkomsel, number of reviews 10 million. As shown in the image below Figure 1.

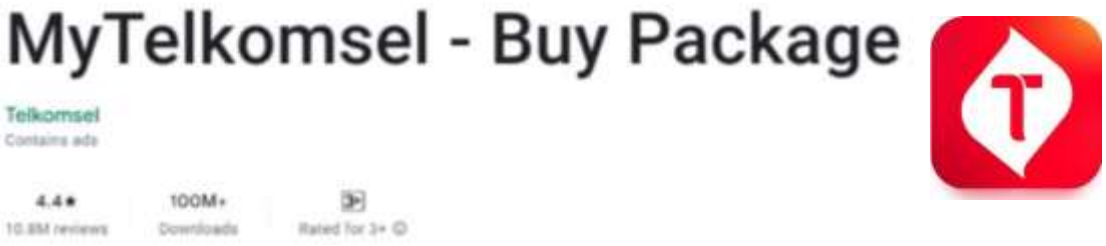

Fig. 1. MyTelkomsel Rating

The use of sentiment analysis techniques in various domains has been the subject of several previous studies; however, this study aims to close this gap by concentrating on mobile applications in Indonesia, specifically the MyTelkomsel application, which has not received much attention in the current literature. Classification with data mining can be done using several methods, namely Support Vector Machine, Decision Tree, K-Nearest Neighbor, Naive Bayes, ID3, CART, Linear Discriminant Analysis and so on, which of course have their respective advantages and disadvantages. [7]. The highest processing dimension was SVM, with 83% accuracy. The understanding dimension achieved a high accuracy of 83% using SVM, while the perception dimension utilized perception and achieved 91% accuracy [8].

The Support Vector Machine (SVM) model, which is effective in handling high-dimensional data and producing accurate classification results, is also used in this work. Support Vector Machines (SVM) is a powerful supervised learning algorithm used for classification or regression [9]. SVM supports both regression and classification tasks and can h andle multiple continuous and categorical variables [10]. SVM is known to perform significantly better than PLS and ANN, with a much lower false positive detection rate [11]. In the context of complex sentiment analysis, the capacity of SVM to manage feature dependencies and map non-linearly splittable data is crucial [12]. Due to its much higher robustness than the former, SVM-based approaches are recommended for practical (industrial) applications [13]. SVM A machine learning approach for making predictions in both classification and regression settings, SVM is a machine learning method.

By using NLP and SVM algorithms, this study attempts to categorize MyTelkomsel program users into two sentiment categories: positive and negative. It is hoped that this work will provide substantial contributions to the creation of sentiment analysis techniques for more relevant and adaptive Indonesian mobile applications.

2. Method

This research was processed in the week of August 3, 2024 with data collection starting in 2017 to July 2024. There are three main stages: the initial stage involves data collection and data pre-processing, followed by the pattern recognition stage, which includes TF-IDF weighting and SVM classification [14]–[16]. Data learning is expected to familiarize researchers with the data that has been collected and can find results regarding what information can be obtained in it [17].

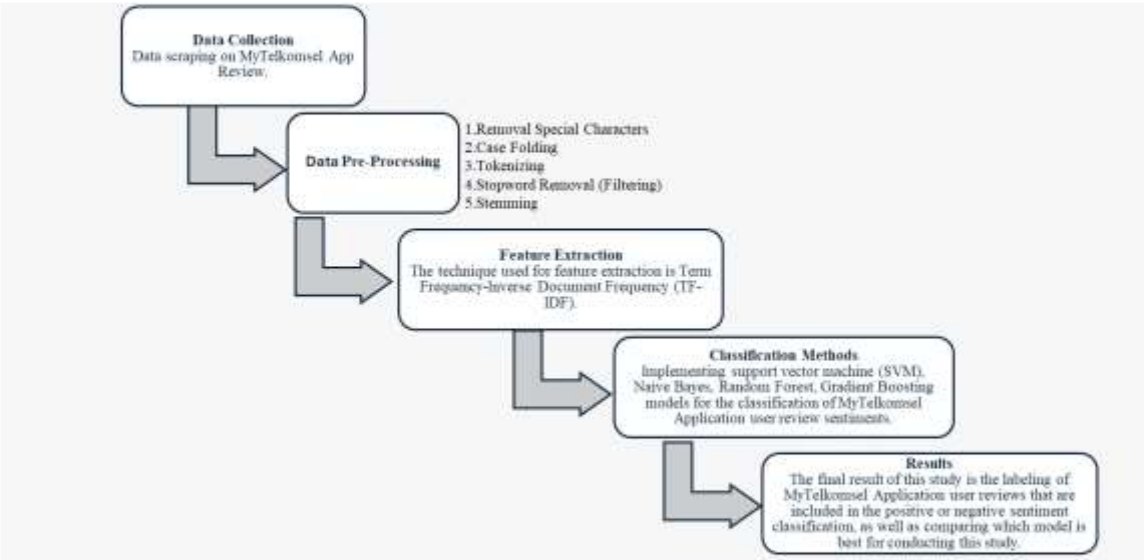

Fig. 2. Research Flow

2.1 Data Collection headings

A critical component of this study is the data collection procedure, which aims to ensure that the information collected is representative of the MyTelkomsel program user population. Google Play Store provides a total of 39.493 reviews data collected using Google Play Scraper, a tool that makes large-scale data collection fast and effective. This data is cleaned and validated to ensure excellent data quality and prevent bias. This data contains various features, including review date, star score, review text content, and user metadata.

2.2 Data Pre-processing

Before the data can be used for sentiment analysis, a series of pre-processing steps are required to transform the raw review text into a more structured format ready for use in a machine learning model. In this process, various specific techniques are systematically applied to achieve the broader goal of improving the accuracy and reliability of the analysis.

Bulleted lists may be included and should look like this:

- Text Cleaning: The first step in pre-processing is cleaning the text from irrelevant elements. Using regular expressions (regex), the review text is cleaned from mentions, hashtags, retweets, links, numbers, and punctuation [18].
- Lowercasing: After the text is cleaned, all letters in the review are converted to lowercase [18]
- Tokenization: This process divides the review text into smaller units called tokens, usually words or short phrases [19].
- Stop Words Removal: To increase the focus of the analysis on meaningful words, common words that do not carry specific information are removed from the text [20].
- Stemming: The final step in pre-processing is stemming, which is the process of converting words to their base form [21].

2.3 Feature Extraction Using TF-IDF

After data pre-processing is complete, the next step is to extract the features that will be used in the machine learning model. The technique chosen for feature extraction is Term Frequency-Inverse Document Frequency (TF-IDF). Feature extraction is converting text documents from any format into a list of features that can be easily processed through text classification techniques [22]. TF-IDF is a weighting system that assigns weight to each word in a document according to the term frequency (tf) and the inverse of the document frequency (idf) [23].

1) TF-IDF Application Process

At this stage, each review is converted into a vector representation using TF-IDF. In this study, the TF-IDF Vectorizer object from the scikit-learn library is used, which is set with certain parameters to ensure optimal feature extraction. For example, max features is set to 200, which means that only the top 200 features based on the TF-IDF value are considered. In addition, min df is set to 17 to ignore words that are too rare, and max df is set to 0.8 to ignore words that are too common in most documents.

2) Conversion to Data Frame

The result of the TF-IDF application is a matrix that shows the weight of each word in each review. This matrix is then converted to a Data Frame using Pandas, which makes it easier to visualize and analyze further. Each row in the Data Frame represents a single review, while its columns represent words with a particular TF-IDF value.

3) Interpretation of TF-IDF Results

A high TF-IDF value indicates that a word occurs frequently in one review but is uncommon across all other reviews. For example, words such as "good," "fast," or "bad" may have high TF-IDF values, reflecting that these words are frequently used in contexts that express sentiment. This process helps in identifying the most significant features that will be used by the machine learning model to determine sentiment.

4) Feature Selection

From the TF-IDF matrix, only features that have significant weight values are selected for further analysis. This approach ensures that the machine learning model is not flooded with less relevant data, which can reduce the prediction accuracy. In this study, the selected features were used to train a Support Vector Machine (SVM) model, which aims to classify sentiment as positive or negative.

2.4 Sentiment Classification Using SVM

After performing data pre-processing and feature extraction using TF-IDF, the next step is to implement a Support Vector Machine (SVM) model for sentiment classification. SVM is a powerful supervised learning algorithm with pattern recognition, and classification [24]–[26]. The goal of document classification is to provide the most appropriate label for a given document .[27].

1) SVM Model Selection

In this study, the SVM model with a linear kernel was chosen because it is simple and often gives good results for text classification problems. This model is implemented using the scikit-learn library. Linear kernel SVM works by finding the optimal hyperplane that separates user review data based on positive and negative sentiments [28]. This process involves using a kernel function to map the data into a higher-dimensional space, allowing for a clearer separation between classes.

2) Model Training

The SVM model is trained using review data that has had its features extracted with TFIDF. In this study, the data was divided into two parts: 90% of the data was used for model training, while the remaining 10% was used for testing. This training process involves adjusting the model parameters to maximize the prediction accuracy on the training data.

3) Model Prediction and Evaluation

Once the SVM model is trained, it is used to predict sentiment on the test dataset. Model evaluation is done by calculating metrics such as precision, recall, and F1-score to provide a comprehensive picture of the model's ability to classify user reviews. The results of this evaluation are important to assess how well the model can distinguish between positive and negative sentiments, and to ensure that the model not only performs well on the training data but is also able to generalize well on data that has never been seen before.

4) Analysis of Evaluation Results

The evaluation results show that the SVM model is able to handle data imbalance and variations in user review texts. The model's ability to predict different sentiments with an acceptable level of accuracy indicates that the model has good generalization capabilities. This means that the model can be relied on to be used on new reviews that have never been seen before.

5) Implementation of Sentiment Prediction on New Data

The trained SVM model is then used to predict sentiment from new sentences entered by the user. Through a similar process involving text cleaning, tokenization, and transformation using TF-

IDF, new sentences can be converted into numeric representations that can be used by the SVM model to predict whether the sentiment of the sentence is positive or negative. The implementation of SVM in this sentiment analysis provides a scalable and reliable approach to identifying user perceptions of the MyTelkomsel application. With a clear method and the use of appropriate evaluation metrics, this model can provide valuable insights for application developers in improving the quality and user experience based on the feedback they receive.

3. Results and Discussion

3.1. Implementation of Support Vector Machine Algorithm

In this study, the Support Vector Machine (SVM) model was used for the task of sentiment classification of MyTelkomsel application reviews. The performance of each model was evaluated using key metrics including precision, recall, F1-score, and accuracy, to determine its effectiveness in identifying positive and negative sentiments.

The assessment findings show that the SVM model produces good results with an accuracy rate of 85.47%, SVM shows that the majority of evaluations classified by this model as positive or negative truly reflect the mood. Furthermore, the recall of the SVM model is close to 85%, indicating its efficacy in identifying all reviews that have a certain sentiment. The harmonic mean of precision and recall, or F1-score, is 85.4%, indicating that the model's performance in sentiment categorization achieves a decent compromise between completeness and accuracy. SVM effectively generalizes well on previously unknown test data, with an overall accuracy of 85.47%. The success of accuracy in applying the Support Vector Machine method is affected by several factors, one of which is the composition of the number of positive and negative data [29].

Table 1. Confusion Matrix for Support Vector Machine

Prediction	Actual	
	Positive	Negative
Positive	2498	576
Negative	571	4254

Table 2. Classification Report Support Vector Machine

Model	Precision	Recall	F1-Score
Positive	81%	81%	81%
Negative	88%	88%	88%

Overall, SVM has a good balance between precision and recall, especially for the negative class. F1-Score also shows a balanced result between the two classes with a value of 0.88 for negative and 0.81 for positive, which means this model provides quite accurate and reliable prediction results.

3.2. Implementation of Naive Bayes Algorithm

Naïve Bayes is one of the most popular data mining algorithms. Its efficiency comes from the assumption of attribute independence, although this might be violated in many real-world data sets. Many efforts have been done to mitigate the assumption, among which attribute selection is an important approach. However, conventional efforts to perform attribute selection in naïve Bayes suffer from heavy computational overhead. This paper proposes an efficient selective naïve Bayes algorithm, which adopts only some of the attributes to construct selective naïve Bayes models. These models are built in such a way that each one is a trivial extension of another. The most predictive selective naïve Bayes model can be selected by the measures of incremental leave-one-out cross validation. As a result, attributes can be selected by efficient model selection. Empirical results demonstrate that the selective naïve Bayes shows superior classification accuracy, yet at the same time maintains the simplicity and efficiency.[30]

This Naive Bayes model shows very good performance in detecting the positive class, with a very high recall (0.94) which means that this model can correctly identify most positive samples. However, this model has weaknesses in detecting the negative class, with a recall of only 0.52, indicating that many negative samples are incorrectly classified as positive (false positive). The precision for the negative class is quite high at 0.84, but the precision for the positive class is lower, at 0.76, meaning that 24% of positive predictions are incorrect (false positive). The F1-Score shows that the positive class has a good balance between precision and recall with a value of 0.84, while the negative class is lower with an F1-Score of 0.64, indicating an imbalance between precision and recall in the negative class. Overall, this model excels in detecting the positive class, but is not as good at detecting the negative class. The test accuracy of 82.97% shows that this model is quite reliable overall, but has shortcomings in handling the negative class which can cause an imbalance in its performance. as shown in table 3 and table 4

Table 3. Confusion Matrix for Naive Bayes

Prediction	**Actual**	
	Positive	Negative
Positive	4254	576
Negative	571	2498

Table 4. Classification Report Naive Bayes

Model	**Precision**	**Recall**	**F1-Score**
Positive	81%	81%	81%
Negative	88%	88%	88%

3.3. Implementation of Random Forest Algorithm

The random forest algorithm can combine the characteristics of multiple eigenvalues, and the combined results of multiple decision trees can be used to improve prediction accuracy. Based on the random forest ensemble learning method, the results of multiple weak classifiers can be combined to produce accurate classification results [31]. The Random Forest technique is a regression tree technique that uses bootstrap aggregation and predictor randomization to achieve a high level of predictive accuracy. Various random forest input parameters are explored [32].

The Random Forest model shows quite good performance, especially in detecting the positive class with high recall (0.89), which means the model is able to detect most of the positive samples. However, the recall for the negative class is lower (0.74), which indicates that this model still produces some errors in predicting the negative class (false positive). The precision for both classes is quite high (0.84 for positive and 0.81 for negative), which means that most of the positive and negative predictions given by this model are correct. The F1-Score for the positive class (0.86) is also higher than the negative class (0.77), which indicates a better performance balance for the positive class. Overall, the Random Forest model has a good performance with an accuracy of 83%, although there is room for improvement especially in detecting the negative class. As shown in table 5 and table 6.

Table 5. Confusion Matrix for Random Forest

Prediction	**Actual**	
	Positive	Negative
Positive	4283	547
Negative	789	2271

Table 6. Classification Report Random Forest

Model	Precision	Recall	F1-Score
Positive	84%	89%	86%
Negative	81%	74%	77%

3.4. Implementation of Gradient Boosting Algorithm

Gradient boosting machines, the learning process successively fits fresh prototypes to offer a more precise approximation of the response parameter [33]. Boosting is an ensemble machine learning technique which combines several low accuracy models to create a high accuracy model [34]. Gradient boosting machines are a family of powerful machine-learning techniques that have shown considerable success in a wide range of practical applications [35].

The Gradient Boosting model shows good performance in detecting the positive class, with a fairly high recall (0.86), which means that this model is able to identify most of the positive samples. However, the recall for the negative class is slightly lower at 0.75, which indicates that the model still produces errors in detecting the negative class (false positive). The precision for the positive (0.84) and negative (0.78) classes is quite adequate, indicating that most of the model's predictions for both classes are correct. The F1-Score for the positive class (0.85) is also higher than the negative class (0.76), indicating better performance in detecting the positive class. Overall, this Gradient Boosting model has an accuracy of 82%, indicating a fairly good performance in completing the classification task, although there is a slight imbalance in detecting the negative class. As shown in table 7 and table 8.

Table 7. Confusion Matrix for Gradient Boosting

Prediction	**Actual**	
	Positive	Negative
Positive	4169	661
Negative	766	2303

Table 8. Classification Report Gradient Boosting

Model	Precision	Recall	F1-Score
Positive	84%	86%	85%
Negative	78%	75%	76%

3.5. Sentiment Distribution

In this study, sentiment analysis was used on 39.493 user evaluations collected from the Google Play Store for the MyTelkomsel application. The attitude of each review was divided into two groups: good and bad. This study did not use the neutral sentiment category because the emphasis was on evaluations with clearly emotional content.

Based on the results of the sentiment analysis, it was determined that 39.2% of user reviews were classified as positive and 60.8% of reviews were not good. This percentage indicates that most of the evaluations received by the MyTelkomsel application were not good. Figure 3 shows a pie chart of user sentiment distribution, which illustrates the distribution of sentiment.

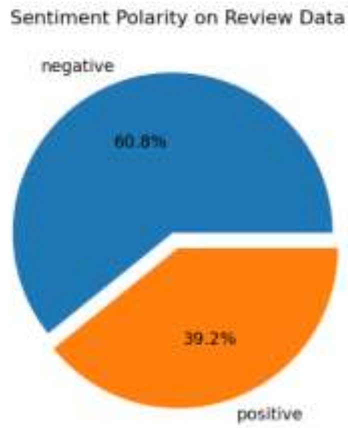

Fig. 3. Distribution of MyTelkomsel user review sentiment

More than half of the reviews were negative, as seen in Figure 2. This indicates that customers had serious issues. The majority of these negative reviews centered on complaints about the service, as well as technological issues such as frequent application failures or glitches. Many users voiced dissatisfaction with the reliability of the application, frequent delays, and navigation challenges. These complaints suggest that technical difficulties were a significant contributor to the negative perception among consumers. In contrast, positive reviews often highlighted characteristics that appealed to customers, such as the application’s speed, usability, and tools for managing their Telkomsel services. Positive user experiences often led users to appreciate the application’s ease of use and effectiveness in meeting their daily needs. Of the 39.493 MyTelkomsel application users, 24.010 user reviews were classified as negative and 15.483 user reviews were classified as positive. This can be seen in Figure 4.

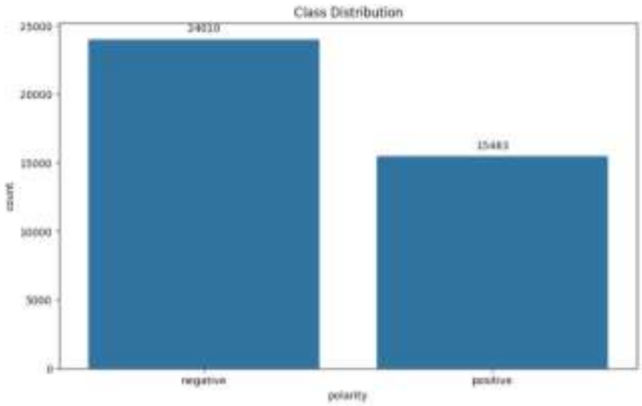

Fig. 4. User Sentiment Distribution Graph

This sentiment distribution offers detailed insights into components that developers need to maintain and areas that could be improved. To improve the overall user experience, developers should prioritize improving the technology and reliability of the application, as more than half of the reviews show negative emotions.

3.6. Keyword Visualization Using Word Cloud

Visualizing keywords using the word cloud technique shows that terms such as “expensive,” “slow,” and “ugly” frequently appear in negative reviews. In contrast, positive reviews often include words such as “fast,” “good,” and “easy,” indicating features that users value. As seen in Figure 5.

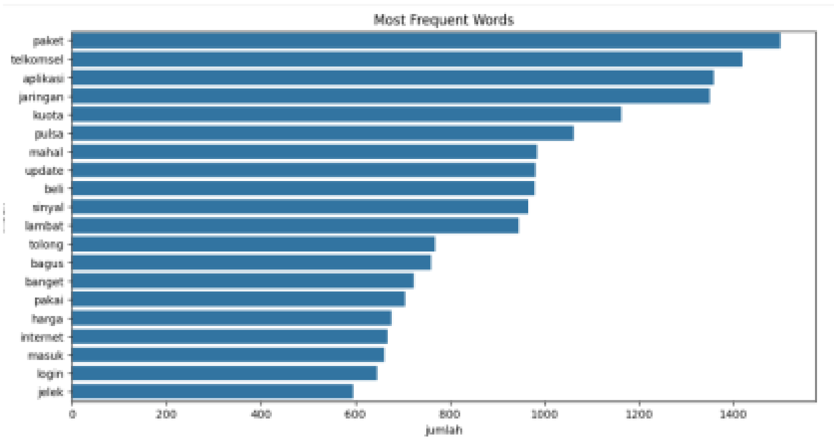

Fig. 5. Number of Words in User Reviews

3.7. Visualization Of Review Data

Visualizing Google Play Store review data using WordCloud helps identify frequently occurring words in text and common sentiments, such as positive, negative, and neutral words.

Fig. 6. Number of Words in User Reviews

3.8. Implications

The results of this sentiment analysis provide several important implications:

1) *Technical Performance Improvement*

Given the high number of negative reviews related to technical issues, developers should focus on improving the stability and reliability of the application.

2) *Service and Pricing Optimization*

Price-related complaints indicate the need to review pricing policies and customer service.

3) *Positive Review-Based Development*

Using insights from positive reviews to improve features that users like can improve overall user satisfaction.

Table 9. Comparation table

Model	Accuracy	Precision	Recall	F1-Score
SVM	0.854792	0.854835	0.854792	0.854813
Naïve Bayes	0.775541	0.787532	0.775541	0.761587
Random Bayes	0.829725	0.828548	0.829725	0.828251
Gradient Bosting	0.819344	0.818443	0.819344	0.818743

Table 1 above compares the performance of four machine learning models, namely SVM (Support Vector Machine), Naive Bayes, Random Forest, and Gradient Boosting, using four main evaluation metrics: accuracy, precision, recall, and F1-Score. From the results shown, SVM ranks first in all metrics, with the highest accuracy (0.854792), precision (0.854835), recall (0.854792), and F1-Score (0.854813). This shows that SVM is very effective in predicting accurately and balances between correct positive detections (recall) and the number of correct positive predictions (precision).

Random Forest is in second place in terms of accuracy (0.827147) and F1-Score (0.828251), which shows quite good performance in maintaining a balance between precision and recall, but still below SVM. Gradient Boosting also shows good performance with accuracy (0.819344) and F1-Score (0.818443) which are close to Random Forest, but still below SVM. On the other hand, Naive Bayes has the lowest performance in all metrics, with accuracy (0.775541), precision (0.787532), recall (0.775541), and F1-Score (0.761587), which shows that this model is less effective than other models in predicting accurately and balanced.

Overall, it can be concluded that SVM is the most superior model, able to provide consistent results both in terms of accuracy and balance between precision and recall, making it suitable for the classification problem faced. Random Forest and Gradient Boosting models can still be good alternatives, while Naive Bayes shows relatively lower performance compared to the others.

4. Conclusion

This study emphasizes the importance of sentiment analysis on MyTelkomsel application user reviews that have been carried out using several machine learning algorithms, namely Support Vector Machine (SVM), Naive Bayes, Random Forest, and Gradient Boosting. From the results of the model performance evaluation, SVM showed the most accurate results with an accuracy rate of 85.47%, making it the most effective method for classifying positive and negative sentiments. Random Forest and Gradient Boosting also showed quite good performance, although not as accurate as SVM. On the other hand, Naive Bayes performed lower than other models.

Furthermore, the analysis revealed that technical issues such as “expensive,” “slow,” and “bad” were the main complaints of users, while speed and ease of use were appreciated by users with positive sentiment. Most of the MyTelkomsel application user reviews were negative, with 60.8% of reviews reflecting user dissatisfaction, especially related to technical issues and application performance. However, the positive reviews that exist highlight the speed and ease of use aspects of the application.

The results of this analysis provide valuable insights for developers to improve the technical quality of the application and improve the overall user experience. However, this study has several limitations. The analysis only uses data from the Google Play Store and may not fully represent all MyTelkomsel app users. In addition, although SVM shows good performance, other methods such as deep learning can be tested to improve the accuracy of sentiment classification. Future research can explore the use of models such as Long Short-Term Memory (LSTM) or Convolutional Neural Networks (CNN) for deeper and adaptive sentiment analysis. With this approach, it is expected that sentiment classification can be more accurate and provide richer insights into user experience.

Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] R. Z. A. Aziz, "The opportunities for MSMEs in the industrial technology," in *Proceeding International Conference on Information Technology and Business*, 2019, hal. 272–286.
[2] F. Fajrin dan N. Andini, "Analisis Kepuasan Pelanggan Ojek Online Di Kawasan Kota Bandung," *J. Ilm. Manaj. Dan Bisnis*, vol. 2, no. 1, hal. 1–13, 2023.
[3] S. Kemp dan others, "Digital 2021: Indonesia." 2021.
[4] M. M. Rahman, S. S. M. M. Rahman, S. M. Allayear, M. F. K. Patwary, dan M. T. A. Munna, "A sentiment analysis based approach for understanding the user satisfaction on android application," in *Data Engineering and Communication Technology: Proceedings of 3rd ICDECT-2K19*, 2020, hal. 397–407.
[5] P. H. Prastyo, I. Ardiyanto, dan R. Hidayat, "Indonesian Sentiment Analysis: An Experimental Study of Four Kernel Functions on SVM Algorithm with TF-IDF," in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*, 2020, hal. 1–6.
[6] F. Susanti, "Sentiment Analysis Provider by. U on Google Play Store Reviews with TF-IDF and Support Vector Machine (SVM) Method," 2020.
[7] P. Riswanto, R. Z. A. Aziz, dan others, "PENERAPAN DECISION TREE C4. 5 SEBAGAI SELEKSI FITUR DAN SUPPORT VECTOR MACHINE (SVM) UNTUK DIAGNOSA KANKER PAYUDARA," *J. Inform.*, vol. 19, no. 1, hal. 54–61, 2019.
[8] M. S. Hasibuan dan R. Z. A. Aziz, "Detection of learning styles with prior knowledge data using the SVM, K-NN and Na{\"\i}ve Bayes algorithms," *J. Infotel*, vol. 14, no. 3, hal. 209–213, 2022.
[9] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.*, vol. 2, no. 2, hal. 121–167, 1998.
[10] O. Chapelle, V. Vapnik, O. Bousquet, dan S. Mukherjee, "Choosing multiple parameters for support vector machines," *Mach. Learn.*, vol. 46, hal. 131–159, 2002.
[11] J. A. F. Pierna, V. Baeten, A. M. Renier, R. P. Cogdill, dan P. Dardenne, "Combination of support vector machines (SVM) and near-infrared (NIR) imaging spectroscopy for the detection of meat and bone meal (MBM) in compound feeds," *J. Chemom. A J. Chemom. Soc.*, vol. 18, no. 7–8, hal. 341–349, 2004.
[12] A. Setyawan, F. Y. Arini, dan I. Akhlis, "Comparative analysis of Simple Additive Weighting method and weighted product method to new employee recruitment Decision Support System (DSS) at PT. Warta Media Nusantara," *Sci. J. Informatics*, vol. 4, no. 1, hal. 34–42, 2017.
[13] R. M. Balabin dan E. I. Lomakina, "Support vector machine regression (SVR/LS-SVM)an alternative to neural networks (ANN) for analytical chemistry? Comparison of nonlinear methods on near infrared (NIR) spectroscopy data," *Analyst*, vol. 136, no. 8, hal. 1703–1712, 2011.
[14] N. P. Husain, S. Sukirman, dan S. SAJIAH, "Analisis Sentimen Ulasan Pengguna Tiktok pada Google Play Store Berbasis TF-IDF dan Support Vector Machine," *J. Syst. Comput. Eng.*, vol. 5, no. 1, hal. 91–102, 2024.

[15] H. Harnelia, "ANALISIS SENTIMEN REVIEW SKINCARE SKINTIFIC DENGAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, 2024.

[16] V. Fitriyana, L. Hakim, D. C. R. Novitasari, dan A. H. Asyhar, "Analisis Sentimen Ulasan Aplikasi Jamsostek Mobile Menggunakan Metode Support Vector Machine," *J. Buana Inform.*, vol. 14, no. 01, hal. 40–49, 2023.

[17] M. S. Hasibuan, R. Z. abdul Aziz, D. Naista, dan N. A. Syafira, "Implementation Of A Classification Algorithm To Detect Felder-Silverman Learning Style."

[18] H. C. Husada dan A. S. Paramita, "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *Teknika*, vol. 10, no. 1, hal. 18–26, 2021.

[19] A. T. J. Harjanta, "Preprocessing text untuk meminimalisir kata yang tidak berarti dalam proses text mining," *J. Inform. Upgris*, vol. 1, no. 1 Juni, 2015.

[20] H. P. P. R. Zuriel dan A. Fahrurozi, "Implementasi Algoritma Klasifikasi Support Vector Machine Untuk Analisa Sentimen Pengguna Twitter Terhadap Kebijakan Psbb," *J. Ilm. Inform. Komput.*, vol. 26, no. 2, hal. 149–162, 2021.

[21] M. F. Al-shufi dan A. Erfina, "Sentimen Analisis Mengenai Aplikasi Streaming Film Menggunakan Algoritma Support Vector Machine Di Play Store," in *Prosiding Seminar Nasional Sistem Informasi dan Manajemen Informatika Universitas Nusa Putra*, 2021, vol. 1, hal. 156–162.

[22] A. I. Kadhim, "Term weighting for feature extraction on Twitter: A comparison between BM25 and TF-IDF," in *2019 international conference on advanced science and engineering (ICOASE)*, 2019, hal. 124–128.

[23] I. Widaningrum, D. Mustikasari, R. Arifin, S. L. Tsaqila, dan D. Fatmawati, "Algoritma Term Frequency--Inverse Document Frequency (TF-IDF) dan K-Means Clustering Untuk Menentukan Kategori Dokumen," *Pros. SISFOTEK*, vol. 6, no. 1, hal. 145–149, 2022.

[24] E. Widjaja, W. Zheng, dan Z. Huang, "Classification of colonic tissues using near-infrared Raman spectroscopy and support vector machines," *Int. J. Oncol.*, vol. 32, no. 3, hal. 653–662, 2008.

[25] V. Vapnik, *The nature of statistical learning theory*. Springer science \& business media, 2013.

[26] M. Tahir, A. Khan, dan A. Majid, "Protein subcellular localization of fluorescence imagery using spatial and transform domain features," *Bioinformatics*, vol. 28, no. 1, hal. 91–97, 2012.

[27] D. Kim, D. Seo, S. Cho, dan P. Kang, "Multi-co-training for document classification using various document representations: TF--IDF, LDA, and Doc2Vec," *Inf. Sci. (Ny).*, vol. 477, hal. 15–29, 2019.

[28] L. Ardiani, H. Sujaini, dan T. Tursina, "Implementasi sentiment analysis tanggapan masyarakat terhadap pembangunan di Kota Pontianak," *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 8, no. 2, hal. 183–190, 2020.

[29] M. A. Syahira dan R. Kurniawan, "Analisis Sentimen Cyberbullying Pada Media Sosial X Menggunakan Metode Support Vector Machine," *J. MEDIA Inform. BUDIDARMA*, vol. 8, no. 3, hal. 1724–1733, 2024.

[30] S. Chen, G. I. Webb, L. Liu, and X. Ma, "A novel selective naïve Bayes algorithm," Knowledge-Based Systems, vol. 192, p. 105361, 2020.

[31] B. Dai, R. C. Chen, S. Z. Zhu, and W. W. Zhang, "Using random forest algorithm for breast cancer diagnosis," in 2018 International Symposium on Computer, Consumer and Control (IS3C), 2018, pp. 449-452.

[32] S. J. Rigatti, "Random forest," Journal of Insurance Medicine, vol. 47, no. 1, pp. 31-39, 2017.

[33] M. S. Hosen and R. Amin, "Significant of gradient boosting algorithm in data management system," Eng. Int., vol. 9, no. 2, pp. 85-100, 2021.

[34] P. Bahad and P. Saxena, "Study of adaboost and gradient boosting algorithms for predictive analytics," in International Conference on Intelligent Computing and Smart Communication 2019: Proceedings of ICSC 2019, Springer, Singapore, 2020, pp. 235-244.

[35] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," Frontiers in Neurorobotics, vol. 7, p. 21, 2013.