Comparative Analysis of Naive Bayes and Support Vector Machine for Sentiment Analysis of Mobile Banking Application Reviews in Indonesia

Yusuf Wisnu Mandaya[1*], Laksamana Rajendra Haidar Azani Fajri [2]
[1,2]Universitas Islam Negeri Sunan Kudus, Jl. Conge Ngembalrejo Bae Kudus Central Java, Kudus and 59322, Indonesia
*Corresponding Author: wisnumandaya@uinsuku.ac.id

ABSTRACT
This study aims to analyze user sentiment toward mobile banking applications in Indonesia using machine learning approaches and to compare the performance of Naive Bayes and Support Vector Machine (SVM) algorithms. The dataset was collected through scraping user reviews from Google Play Store, focusing on three major mobile banking applications: BCA Mobile, Livin by Mandiri, and BRImo. The research methodology includes data preprocessing (case folding, cleaning, and stemming), feature extraction using Term Frequency–Inverse Document Frequency (TF-IDF), optional feature selection using Chi-Square, and classification using Naive Bayes and SVM. The performance of each model was evaluated using accuracy, precision, recall, and F1-score metrics. The experimental results show that the SVM algorithm achieved the highest accuracy of 89.80%, outperforming Naive Bayes, which achieved 86.22%. Meanwhile, the application of Chi-Square feature selection did not significantly improve model performance and slightly reduced accuracy to 85.71%. These findings indicate that SVM is more effective in handling high-dimensional text data for sentiment classification tasks, while TF-IDF representation alone is sufficient without additional feature selection. This study contributes to the evaluation of machine learning methods for sentiment analysis in mobile banking applications and provides insights for improving digital banking services based on user feedback.

ARTICLE HISTORY
Received 28 April 2026
Revision 11 May 2026
Accepted 13 May 2026

KEYWORD
Mobile Banking;
Naive Bayes;
Sentiment Analysis;
Support Vector Machine;
Text Mining

1. INTRODUCTION

The rapid advancement of information technology has significantly transformed the banking industry, particularly through the development of digital financial services such as mobile banking applications. In Indonesia, mobile banking platforms including BCA Mobile, Livin by Mandiri, and BRImo have become essential tools that enable users to conduct financial transactions efficiently without visiting physical bank branches. The increasing adoption of mobile banking applications is driven by the growing demand for convenience, transaction speed, and accessibility in digital financial services (Baabdullah et al., 2019).

Along with the increasing use of mobile banking services, users actively provide feedback and opinions through digital platforms such as Google Play Store. These reviews contain valuable information regarding user experiences, service quality, application performance, system reliability, and security issues. Positive reviews generally indicate user satisfaction, while negative reviews often represent complaints related to application errors, transaction failures, login problems, and system instability. Such information is important for financial institutions to evaluate service quality and improve customer satisfaction. However, the continuously increasing volume of user reviews makes manual analysis inefficient, time-consuming, and impractical (B. Liu, 2014).

To address this issue, sentiment analysis has emerged as one of the most widely used techniques in text mining and natural language processing for automatically identifying opinions and classifying sentiments expressed in textual data. Sentiment analysis enables organizations to extract meaningful insights from large-scale user-generated content and supports data-driven decision making (Johri et al., 2021; Pang & Lee, 2008). In the context of mobile banking, sentiment analysis can help financial institutions understand customer perceptions and identify critical issues that may affect user trust and service adoption.

Various machine learning methods have been applied in sentiment analysis research. Among these methods, Naive Bayes is one of the most commonly used algorithms due to its simplicity, computational efficiency, and effectiveness in text classification tasks involving sparse textual features (Hutto & Gilbert, 2014; Jeetendra Vadher, 2025; Kamath et al., 2025; Kowsari et al., 2019; Sebastiani, 2002; H. Zhang, 2004). Naive Bayes performs well in handling high-dimensional data generated through feature extraction techniques such as TF-IDF while requiring relatively low computational resources. In addition, the algorithm is widely recognized as a strong baseline model in sentiment classification studies (Rennie et al., 2003). Despite these advantages, Naive Bayes relies on the assumption that features are conditionally

independent, which may reduce its capability to capture complex relationships between textual features in real-world datasets (Sebastiani, 2002).

On the other hand, Support Vector Machine (SVM) has been recognized as one of the most effective supervised learning algorithms for text classification and sentiment analysis. SVM is capable of handling high-dimensional and sparse data by constructing optimal hyperplanes that maximize the separation margin between classes (Cortes et al., 1995). Previous studies have shown that SVM frequently outperforms probabilistic classifiers in sentiment analysis tasks because of its ability to model complex feature distributions and improve classification generalization (Joachims, 1998; Lestari & Hutagalung, 2025; Singh & Sharma, 2023; Subedi et al., 2025). Therefore, comparing Naive Bayes and SVM provides important insights into the effectiveness of probabilistic and margin-based classification approaches in mobile banking sentiment analysis.

In addition to classification algorithms, feature representation and feature selection also play important roles in text classification performance. TF-IDF is widely used as a feature extraction method because it effectively measures the importance of words within documents while reducing the influence of common terms (J Ramos, 2003). Furthermore, feature selection techniques such as Chi-Square are often applied to reduce feature dimensionality and improve model performance by selecting the most relevant terms (Forman, 2003). However, the effectiveness of Chi-Square feature selection varies depending on dataset characteristics and classification algorithms. In some cases, feature selection may improve performance, while in other cases it may remove informative features and reduce classification accuracy (Nurhayati et al., 2019).

Several previous studies have explored sentiment analysis using machine learning techniques in domains such as e-commerce, social media, tourism, and online product reviews (J. S. Liu & Tsaur, 2014; Moraes et al., 2013). However, studies specifically focusing on sentiment analysis of mobile banking application reviews in Indonesia remain relatively limited. Moreover, most previous studies only focused on implementing a single classification algorithm without conducting comparative analysis between probabilistic classifiers such as Naive Bayes and margin-based classifiers such as SVM. In addition, the impact of Chi-Square feature selection on sentiment classification performance in the context of Indonesian mobile banking reviews has not been extensively investigated.

Based on these limitations, a research gap can be identified in three aspects. First, limited studies focus specifically on sentiment analysis of Indonesian mobile banking applications using real-world review data collected from multiple banking platforms. Second, comparative evaluation between Naive Bayes and SVM in this

domain is still limited. Third, the effectiveness of Chi-Square feature selection in improving sentiment classification performance for Indonesian mobile banking reviews has not been comprehensively analyzed.

Therefore, this study aims to analyze user sentiment toward mobile banking applications in Indonesia using Naive Bayes and Support Vector Machine algorithms. This study also evaluates the impact of Chi-Square feature selection on classification performance using review data collected from BCA Mobile, Livin by Mandiri, and BRImo applications. The novelty of this research lies in the comparative evaluation of probabilistic and margin-based classification approaches for Indonesian mobile banking sentiment analysis, as well as the investigation of the effectiveness of Chi-Square feature selection using real-world mobile banking review datasets. The findings of this study are expected to contribute both theoretically and practically by providing insights into suitable machine learning approaches for sentiment analysis and supporting financial institutions in improving digital banking service quality based on user feedback.

2. The Proposed Method

2.1 Sentiment Analysis

Sentiment analysis is a text mining technique used to identify and classify opinions expressed in textual data. It is widely applied in analyzing user-generated content such as online reviews, social media posts, and feedback systems. In this study, sentiment analysis is used to classify mobile banking user reviews into positive and negative categories to understand user satisfaction and system performance (B. Liu, 2014).

2.2 Naïve Bayes

Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem. It assumes that features are independent of each other given the class label. Due to its simplicity and efficiency, Naive Bayes is widely used in text classification tasks, including sentiment analysis. However, its independence assumption may limit its ability to capture complex relationships between features in textual data (L. Zhang, 2025).

2.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane that separates data into different classes with maximum margin. SVM is particularly effective for high-dimensional data such as text, as it can handle complex patterns and improve classification accuracy compared to simpler models (Joachims, n.d.).

2.4 TF-IDF (Term Frequency–Inverse Document Frequency)

TF-IDF is a feature extraction technique used to convert textual data into numerical representations. It assigns weights to words based on their frequency in a document and their importance across the dataset. This method helps highlight relevant terms while reducing the influence of common words, making it suitable for sentiment classification (J Ramos, 2003; Lestari & Hutagalung, 2025).

2.5 Chi-Square Feature Selection

Chi-Square is a statistical method used to evaluate the relationship between features and class labels. It is commonly applied in text classification to reduce feature dimensionality by selecting the most relevant terms. In this study, Chi-Square is used as an optional feature selection technique to evaluate its impact on model performance (Nurhayati et al., 2019).

2.6 Mobile Banking Applications

Mobile banking applications allow users to perform financial transactions digitally. Applications such as BCA Mobile, Livin by Mandiri, and BRImo are widely used in Indonesia. User reviews of these applications provide valuable insights into system usability, performance, and service quality.

3. RESEARCH METHODS

This study employed an experimental quantitative approach to classify user sentiment toward mobile banking applications in Indonesia. The research was conducted by collecting user review data from Google Play Store, preprocessing textual data, transforming text into numerical features, applying classification algorithms, and evaluating model performance (Aggarwal et al., 2013). The overall research workflow is illustrated in Figure 1.

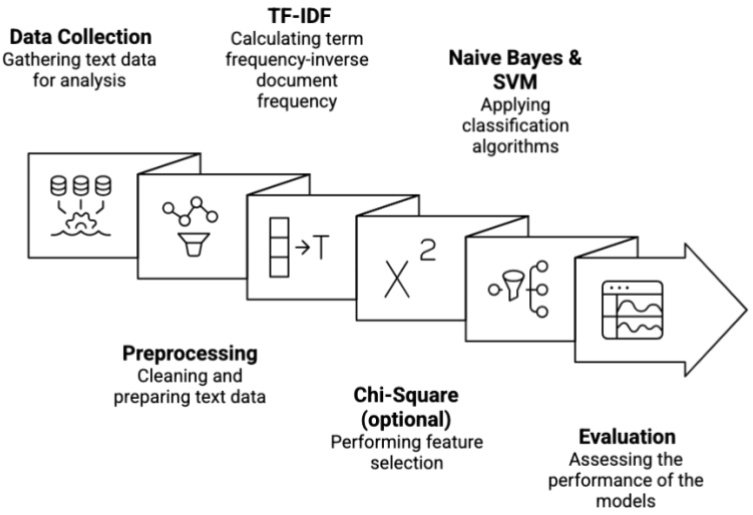

Figure 1. Research methodology of sentiment analysis using TF-IDF, Chi-Square, Naive Bayes, and Support Vector Machine

Figure 1 illustrates the overall research workflow. The process begins with data collection from Google Play Store, followed by preprocessing to clean and standardize the textual data. The data are then transformed using TF-IDF to generate numerical features. Feature selection using Chi-Square is applied as an optional step to evaluate its impact. Finally, classification is performed using Naive Bayes and Support Vector Machine, and the results are evaluated using standard performance metrics.

The preprocessing stage includes cleaning, tokenization, and stemming to prepare the dataset (Jurafsky & Martin, 2021) . Feature extraction is performed using TF-IDF to transform textual data into numerical form (Ramos, 2003).

Classification is performed using Naive Bayes and Support Vector Machine (SVM), which are widely used in text classification tasks (Kowsari et al., 2019). Model evaluation is conducted using accuracy, precision, recall, and F1-score.

3.1 Data Collection

The dataset used in this study was obtained from user reviews on the Google Play Store. Three mobile banking applications in Indonesia were selected as research objects, namely BCA Mobile, Livin by Mandiri, and BRImo. These applications were selected because they are among the widely used mobile banking platforms in Indonesia and represent major digital banking services used by the public.

The data collection process was conducted using a scraping technique. The collected attributes included the review text and rating score given by users. The review text was used as the main input data for sentiment classification, while the rating score was used to generate sentiment labels. Reviews were collected from each application and then combined into a single dataset for further processing.

The use of Google Play Store reviews is considered appropriate because the reviews represent direct user opinions regarding application performance, service quality, transaction experience, and technical issues encountered when using mobile banking services.

3.2 Sentiment Labelling

After the review data were collected, the rating score was converted into sentiment labels. In this study, the sentiment classes were divided into two categories: positive and negative. Reviews with rating scores of 1 and 2 were labelled as negative sentiment, while reviews with rating scores of 4 and 5 were labelled as positive sentiment.

Reviews with a rating score of 3 were excluded from the dataset because they may represent neutral or ambiguous opinions. Removing neutral reviews was intended to reduce ambiguity and improve the clarity of the classification task. Therefore, the final dataset used in the classification process only contained positive and negative sentiment classes. The sentiment labelling rule used in this study is shown below:

Table 1. The sentiment labelling rule

Rating Score	Sentiment Label
1–2	Negative
3	Removed / Neutral

Rating Score	Sentiment Label
4–5	Positive

3.3 Data Balancing

Before the classification process, the distribution of sentiment labels was examined. In sentiment classification tasks, imbalanced data can cause the model to be biased toward the majority class. For example, if negative reviews dominate the dataset, the model may tend to predict most reviews as negative, resulting in misleading accuracy values.

To avoid this issue, the dataset was balanced by equalizing the number of positive and negative reviews. The balancing process was performed by selecting the same number of samples from each sentiment class. This step ensured that both classes had equal representation during model training and testing. By using a balanced dataset, the evaluation results are expected to reflect the actual classification capability of the models rather than the effect of class imbalance.

3.4 Data Preprocessing

Data preprocessing was carried out to clean and standardize the review text before feature extraction. Since user reviews on Google Play Store often contain informal language, punctuation, numbers, emojis, repeated characters, and irrelevant symbols, preprocessing is an important step to improve data quality. The preprocessing stages applied in this study include:

3.4.1 Case Folding

All review text was converted into lowercase letters. This step was performed to avoid different interpretations of the same word caused by capitalization differences. For example, the words “Aplikasi”, “aplikasi”, and “APLIKASI” were treated as the same term.

3.4.2 Cleaning

The cleaning process was conducted to remove irrelevant characters such as numbers, punctuation marks, symbols, and special characters. This step ensured that only textual information was retained for analysis.

3.4.3 Tokenization

Tokenization was performed to split each sentence into individual words or tokens. This process helps the system identify important terms contained in each review.

3.4.4 Stopword Removal

Common words that do not carry significant meaning were removed from the text. Examples of stopwords include words such as “yang”, “dan”, “di”, and “ke”. Removing stopwords helps reduce noise in the dataset.

3.4.5 Stemming

Stemming was applied to convert words into their root forms. For Indonesian text, stemming is important because words may appear in various forms due to prefixes, suffixes, or infixes. For example, the words “menggunakan”, “digunakan”, and

"penggunaan" can be reduced to their root form "guna". After preprocessing, the review text became cleaner and more suitable for feature extraction.

3.5 Feature Extraction Using TF-IDF

After the preprocessing stage, the text data were transformed into numerical features using the Term Frequency–Inverse Document Frequency (TF-IDF) method. Machine learning algorithms cannot directly process raw text; therefore, the text must be represented in numerical form.

TF-IDF assigns weights to words based on their importance in a document relative to the entire dataset. Words that appear frequently in a specific review but are not too common across all reviews receive higher weights. In contrast, words that appear frequently in almost all reviews receive lower weights because they are considered less informative.

In this study, TF-IDF was used to generate feature vectors from the cleaned review text. These feature vectors were then used as input for the classification models.

3.6 Feature Selection using Chi-Square

In addition to TF-IDF, this study also applied Chi-Square feature selection to evaluate whether selecting the most relevant features could improve classification performance. The Chi-Square method measures the dependency between each feature and the sentiment class label.

Features with higher Chi-Square scores are considered more relevant for distinguishing between positive and negative sentiment. By selecting only the most relevant features, the model may reduce noise and improve classification efficiency.

However, in this study, Chi-Square was treated as an additional experimental scenario rather than the main classification method. The purpose was to compare the performance of Naive Bayes with and without feature selection. This comparison was conducted to determine whether Chi-Square feature selection provides a significant improvement in sentiment classification performance.

3.7 Classification Models

This study used two machine learning algorithms for sentiment classification: Naive Bayes and Support Vector Machine (SVM).

3.7.1 Naive Bayes Classification

Naive Bayes was used as the baseline classification model. This algorithm applies a probabilistic approach based on Bayes' theorem and assumes that each feature contributes independently to the classification result.

Naive Bayes was selected because it is computationally efficient, simple to implement, and widely used in text classification tasks. It is particularly suitable for sparse and high-dimensional data produced by TF-IDF representation. In this study, Naive Bayes was tested in two scenarios:

- Naive Bayes using TF-IDF features without Chi-Square feature selection
- Naive Bayes using TF-IDF features with Chi-Square feature selection

These scenarios were used to evaluate whether Chi-Square feature selection improves Naive Bayes classification performance.

3.7.2 Support Vector Machine Classification

Support Vector Machine was used as a comparative classification model. SVM works by finding the optimal hyperplane that separates sentiment classes with the maximum margin. This characteristic makes SVM effective for high-dimensional text data.

SVM was selected because it is widely recognized as a strong classifier for text classification and sentiment analysis. Unlike Naive Bayes, SVM does not rely on the assumption of feature independence, allowing it to better handle complex relationships among textual features.

In this study, SVM was trained using TF-IDF features and compared with the Naive Bayes models to determine the best-performing classification approach.

3.8 Training and Testing Process

The dataset was divided into training and testing data using a train-test split approach. The training data were used to build the classification models, while the testing data were used to evaluate model performance on unseen data.

To maintain the balance of sentiment classes in both training and testing sets, stratified sampling was applied. This ensured that the proportion of positive and negative reviews remained consistent in both subsets. The use of stratified splitting is important because it prevents class distribution imbalance between training and testing data, which could affect the reliability of evaluation results.

3.9 Model Evaluation

Model performance was evaluated using four evaluation metrics: accuracy, precision, recall, and F1-score. Accuracy measures the proportion of correctly classified reviews compared to the total number of reviews. Precision measures how many reviews predicted as a certain sentiment were actually correct. Recall measures how many actual sentiment reviews were successfully identified by the model. F1-score represents the harmonic mean of precision and recall, providing a balanced measure of classification performance. The evaluation metrics used in this study are described as follows:

Table 3. The evaluation metrics

Metric	Description
Accuracy	Measures the overall correctness of classification results
Precision	Measures the correctness of positive or negative predictions
Recall	Measures the ability of the model to identify actual sentiment classes
F1-Score	Measures the balance between precision and recall

In addition to these metrics, a confusion matrix was also used to analyze the classification results in more detail. The confusion matrix shows the number of correctly and incorrectly classified reviews for each sentiment class.

3.10 Experimental Scenarios

To obtain a comprehensive comparison, this study conducted three experimental scenarios:

Table 4. experimental scenarios

Scenario	Model	Feature Representation	Feature Selection
Scenario 1	Naive Bayes	TF-IDF	No
Scenario 2	Naive Bayes	TF-IDF	Chi-Square
Scenario 3	SVM	TF-IDF	No

The first scenario was used to evaluate the baseline performance of Naive Bayes. The second scenario was used to examine the effect of Chi-Square feature selection on Naive Bayes performance. The third scenario was used to compare Naive Bayes with SVM as a margin-based classifier.

Through these scenarios, this study evaluates not only which model produces the highest accuracy, but also whether feature selection contributes to better sentiment classification performance.

3.11 Research Workflow Summary

Overall, the research process began with collecting user reviews from Google Play Store. The rating scores were then converted into sentiment labels, and neutral reviews were removed. The dataset was balanced to ensure equal representation of positive and negative sentiment classes. After that, the text data were preprocessed through case folding, cleaning, tokenization, stopword removal, and stemming.

The cleaned text was transformed into numerical features using TF-IDF. Chi-Square feature selection was then applied in one experimental scenario to evaluate its effect on classification performance. Finally, Naive Bayes and SVM models were trained and tested, and their performance was compared using accuracy, precision, recall, F1-score, and confusion matrix.

This methodological design allows the study to systematically evaluate the effectiveness of different classification approaches for sentiment analysis of mobile banking application reviews in Indonesia.

4. Research Instrument

This section presents the results of sentiment classification using Naive Bayes, Naive Bayes with Chi-Square feature selection, and Support Vector Machine (SVM). The evaluation focuses on comparing model performance using accuracy, precision, recall, and F1-score metrics.

4.1 Model Performance Comparison

The classification results of all models are presented in Table 1.

Table 5. Model Performance Comparison

Model	Accuracy (%)	Precision	Recall	F1-Score
Naïve Bayes	86.22	0.88	0.86	0.86
NB + Chi-Square	85.71	0.88	0.86	0.85
Support Vector Machine (SVM)	**89.80**	**0.91**	**0.90**	**0.90**

Based on Table 1, the experimental results show that SVM achieved the highest accuracy of 89.80%, outperforming Naive Bayes, which achieved 86.22%. This finding is consistent with previous studies that highlight the effectiveness of SVM in handling high-dimensional text data (Singh & Sharma, 2023).

Naive Bayes also performed well but slightly lower due to its independence assumption between features (L. Zhang, 2025). The addition of Chi-Square feature selection did not improve performance, indicating that TF-IDF representation is already sufficient for this dataset (Nurhayati et al., 2019).

These findings confirm that selecting an appropriate classification algorithm has a greater impact on performance than applying feature selection techniques alone (Kowsari et al., 2019).

4.2 Visualization of Model Performance

The comparison of classification accuracy is illustrated in Figure 2.

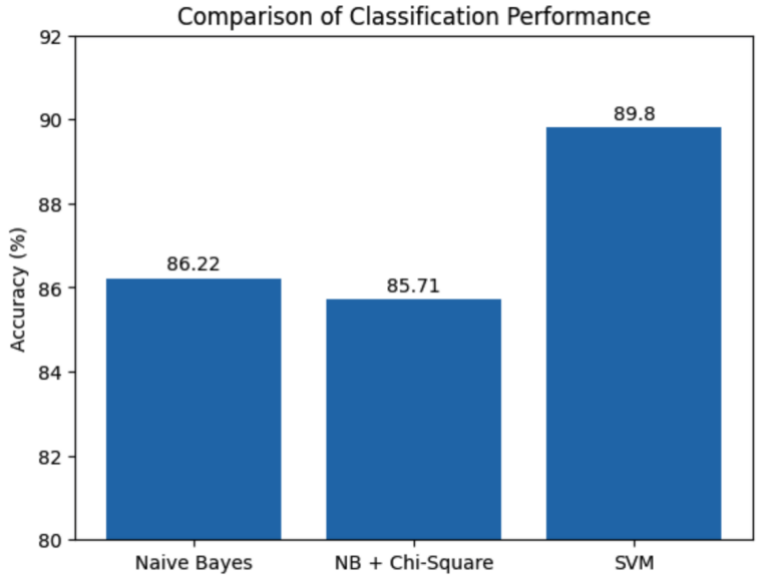

Figure 2. Comparison of classification accuracy between Naïve Bayes, NB with Chi-Square, and SVM

The figure clearly shows that SVM outperforms the other models, confirming its superiority in this study.

4.3 Analysis of SVM Classification Performance

To further evaluate the best-performing model, the detailed classification results of SVM are presented in Table 2.

Table 2. SVM Classification Performance by Class

Class	Precision	Recall	F1-Score	Support
Negative	0.85	0.97	0.90	98
Positive	0.96	0.83	0.89	98

From Table 2, it can be observed that the SVM model demonstrates strong classification capability for both classes.

The model achieved a recall of 0.97 for the negative class, indicating that most negative reviews were correctly identified. This is particularly important in sentiment analysis, as negative reviews often represent user complaints and system issues that require attention.

For the positive class, the model achieved a precision of 0.96, indicating that positive predictions are highly reliable. However, the recall for positive sentiment is slightly lower (0.83), suggesting that some positive reviews were misclassified.

4.4 Confusion Matrix Analysis

The confusion matrix of the SVM model is shown in Figure 3.

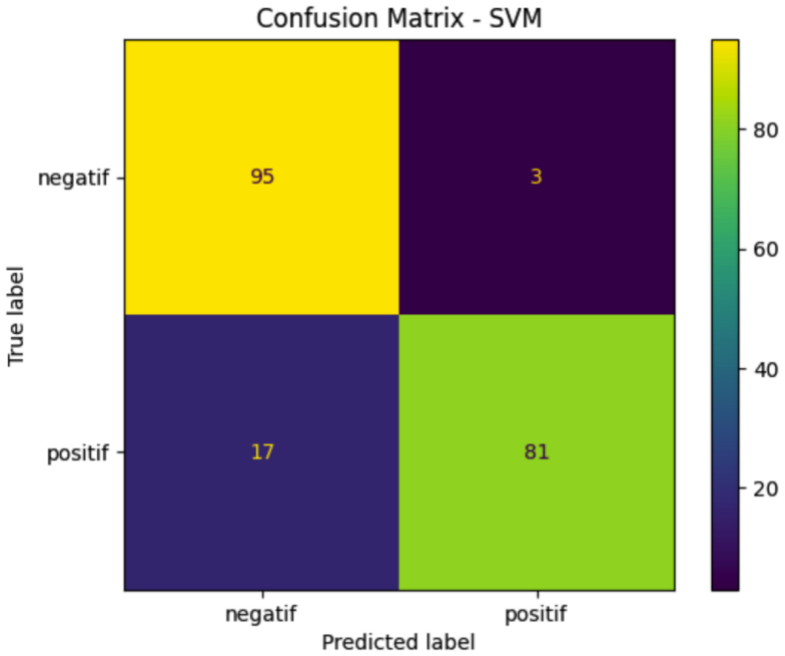

Figure 3. Confusion matrix of SVM classification results

The confusion matrix shows that most data points are correctly classified, with only a small number of misclassifications between positive and negative classes. This indicates that the model performs well in distinguishing sentiment categories.

4.5 Effect of Chi-Square Feature Selection

The experimental results indicate that the application of Chi-Square feature selection does not improve classification performance. Instead, it slightly decreases the accuracy from 86.22% to 85.71%.

This phenomenon may occur because the TF-IDF representation already captures the most important features in the dataset. Therefore, additional feature selection may remove useful information and reduce model performance.

4.6 Discussion

The results of this study confirm that Support Vector Machine is more effective than Naive Bayes for sentiment analysis of mobile banking reviews. This is consistent with previous studies that highlight the ability of SVM to handle high-dimensional and sparse data.

In contrast, Naive Bayes relies on the independence assumption between features, which may not hold in real-world textual data. This limitation can reduce its performance compared to more advanced models such as SVM.

Furthermore, the findings suggest that feature selection techniques such as Chi-Square should be applied carefully, as they may not always improve model performance depending on the dataset characteristics.

Overall, this study demonstrates that selecting an appropriate classification algorithm has a greater impact on performance than applying additional feature selection techniques.

5. CONCLUSION

This study demonstrates that SVM outperforms Naive Bayes in sentiment classification of mobile banking reviews. The results are consistent with previous research showing the superiority of SVM in text classification tasks (Singh & Sharma, 2023).

The experimental results demonstrate that the Support Vector Machine (SVM) algorithm achieved the highest performance with an accuracy of 89.80%, outperforming Naive Bayes, which achieved 86.22%. This indicates that SVM is more effective in handling high-dimensional textual data and capturing complex patterns in sentiment classification tasks.

In addition, the application of Chi-Square feature selection did not significantly improve classification performance and slightly reduced accuracy. This finding suggests that TF-IDF representation alone is sufficient for this dataset, and additional feature selection may not always provide benefits depending on data characteristics (J Ramos, 2003).

From a practical perspective, the results of this study provide valuable insights for analyzing user feedback in mobile banking applications. The ability to accurately classify user sentiment can help financial service providers identify user satisfaction levels, detect system issues, and improve service quality based on user reviews.

The main contribution of this research lies in the comparative evaluation of Naive Bayes and Support Vector Machine for sentiment analysis in the context of mobile banking applications in Indonesia, as well as the assessment of Chi-Square feature selection on model performance. The findings highlight the importance of selecting appropriate classification algorithms rather than relying solely on feature selection techniques.

For future research, it is recommended to explore more advanced methods such as deep learning models, including Long Short-Term Memory (LSTM) and transformer-based approaches, as well as incorporating larger and more diverse datasets. Additionally, aspect-based sentiment analysis can be considered to provide more detailed insights into specific features of mobile banking applications.

6. REFERENCES

Aggarwal, K. K., Jaggi, C. K., & Kumar, A. (2013). An Inventory Decision Model When Demand Follows Innovation Diffusion Process under Effect of Technological Substitution. Advances in Decision Sciences, 2013. https://doi.org/10.1155/2013/915657

Baabdullah, A. M., Alalwan, A. A., Rana, N. P., Kizgin, H., & Patil, P. (2019). Consumer use of mobile banking (M-Banking) in Saudi Arabia: Towards an integrated model. International Journal of Information Management, 44, 38–52. https://doi.org/10.1016/J.IJINFOMGT.2018.09.002

Cortes, C., Vapnik, V., & Saitta, L. (1995). Support-vector networks. Machine Learning 1995 20:3, 20(3), 273–297. https://doi.org/10.1007/BF00994018

Sebastiani, F. (2002). Machine learning in automated text categorization. Dl.Acm.OrgF SebastianiACM Computing Surveys (CSUR), 2002•dl.Acm.Org, 34(1), 1–47. https://doi.org/10.1145/505282.505283

Forman, G. (2003). An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research, 3, 1289–1305.

Hutto, C. J., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1), 216–225. https://doi.org/10.1609/ICWSM.V8I1.14550

J Ramos. (2003). Using tf-idf to determine word relevance in document queries. Proceedings of the First Instructional Conference on Machine Learning, 2003•Citeseer. https://citeseerx.ist.psu.edu/document?repid=rep1type=pdf&doi=b3bf6373ff41a115197cb5b30e57830c16130c2c

Jeetendra Vadher, S. (2025). Performance Analysis of Naïve Bayes and Stochastic Gradient Descent-based SVM for Sentiment Analysis. SSRG International

Journal of Computer Science and Engineering, 12, 10–18. https://doi.org/10.14445/23488387/IJCSE-V12I5P102

Joachims, T. (n.d.). Text Categorization with Support Vector Machines: Learning with Many Relevant Features.

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. 137–142. https://doi.org/10.1007/BFB0026683

Johri, P., Khatri, S. K., Al-Taani, A. T., Sabharwal, M., Suvanov, S., & Kumar, A. (2021). Natural Language Processing: History, Evolution, Application, and Future Work. Lecture Notes in Networks and Systems, 167, 365–375. https://doi.org/10.1007/978-981-15-9712-1_31/SAVE-RESEARCH

Kamath, S., Narendra, V. G., & Shivaprasad, G. (2025). Comparative Analysis of Sentiment Analysis Techniques Employing Machine Learning. 129–138. https://doi.org/10.1007/978-981-96-1452-3_10

Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text Classification Algorithms: A Survey. Information 2019, Vol. 10, Page 150, 10(4), 150. https://doi.org/10.3390/INFO10040150

Lestari, V. B., & Hutagalung, C. A. (2025). Evaluation of TF-IDF Extraction Techniques in Sentiment Analysis of Indonesian-Language Marketplaces Using SVM, Logistic Regression, and Naive Bayes. J-KOMA : Jurnal Ilmu Komputer Dan Aplikasi, 8(1), 36–44. https://doi.org/10.21009/J-KOMA.V8I1.05

Liu, B. (2014). Sentiment Analysis Essentials Sentiment Analysis: Mining Opinions, Sentiments, and Emotions.

Liu, J. S., & Tsaur, S. H. (2014). We are in the same boat: Tourist citizenship behaviors. Tourism Management, 42, 88–100. https://doi.org/10.1016/J.TOURMAN.2013.11.001

Moraes, R., Valiati, J. F., & Gavião Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2), 621–633. https://doi.org/10.1016/J.ESWA.2012.07.059

Nurhayati, Putra, A. E., Wardhani, L. K., & Busman. (2019). Chi-Square Feature Selection Effect on Naive Bayes Classifier Algorithm Performance for Sentiment Analysis Document. 2019 7th International Conference on Cyber and IT Service Management, CITSM 2019. https://doi.org/10.1109/CITSM47753.2019.8965332

Pang, B., & Lee, L. (2008). Opinion Mining and Sentiment Analysis. Foundations and Trends R in Information Retrieval, 2(2), 1–135. https://doi.org/10.1561/1500000001

Rennie, J. D. M., Shih, L., Teevan, J., & Karger, D. R. (2003). Tackling the Poor Assumptions of Naive Bayes Text Classifiers. Proceedings of the Twentieth International Conference on Machine Learning. https://people.csail.mit.edu/jrennie/papers/icml03-nb.pdf

Sebastiani, F. (2002). Machine learning in automated text categorization. ACM Computing Surveys (CSUR), 34(1), 1–47. https://doi.org/10.1145/505282.505283

Singh, J., & Sharma, G. (2023). Sentiment Analysis Study of Human Thoughts using Machine Learning Techniques. 2023 International Conference on Disruptive Technologies, ICDT 2023, 776–785. https://doi.org/10.1109/ICDT57929.2023.10150917

Subedi, D., Lamichhane, N., & Subedi, N. (2025). Sentiment Analysis of IMDb Movie Reviews Using SVM and Naive Bayes Classifier. Journal of Engineering and Sciences, 4(1), 56–68. https://doi.org/10.3126/JES2.V4I1.70138

Zhang, H. (2004). The Optimality of Naive Bayes. American Association for Artificial Intelligence. www.aaai.org

Zhang, L. (2025). Features extraction based on Naive Bayes algorithm and TF-IDF for news classification. PLOS ONE, 20(7), e0327347. https://doi.org/10.1371/JOURNAL.PONE.0327347