Jurnal Ilmiah Informatika dan Komputer (Informatech)
Volume 3, Number 2026, June 2026, page 20-25
DOI : 10.69533
Website : *https://ejournal.rizaniamedia.com/index.php/informatech*

E-ISSN: 3047-4752

Comparative Analysis of Naive Bayes and Support Vector Machine for Sentiment Classification of Indonesian-Language Mobile Application Reviews on Google Play Store

Norris Elden Salassa[1*], Arpen Patanduk[2], Ade Yusupa[3], Yaulie Deo Y. Rindengan[4]

[*1,2,3,4)] Sam Ratulangi University, Bahu, Malalayang District, Manado City, North Sulawesi, Sulawesi Utara, 95115

[1]*email : norrissalassa026@student.unsrat.ac.id*
[2]*email: arpenpatanduk026@student.unsrat.ac.id*
[3]*email: ade@unsrat.ac.id*
[4]*email: rindengan@unsrat.ac.id*

(*Article Received: 27 April 2026; Article Revised: 11 May 2026; Article Published: 1 June 2026;*)

ABSTRACT *– This study conducted a comparative performance evaluation of Multinomial Naive Bayes and Support Vector Machine (SVM) with a linear kernel in classifying the sentiment of Indonesian-language mobile application reviews collected from the Google Play Store. A total of 2,847 reviews targeting the GoPay digital wallet application were gathered via web scraping using the google-play-scraper library. After preprocessing, including case folding, cleansing, tokenization, stopword removal, and stemming using the Sastrawi library, the final dataset comprised 2,634 usable reviews. Sentiment labeling was conducted automatically based on star ratings: ratings of 4 and 5 were assigned as positive (1,841 reviews, 69.9%), while ratings of 1 and 2 were assigned as negative (793 reviews, 30.1%). Feature extraction used TF-IDF with a vocabulary size of 8,432 unique terms. Model training used an 80:20 train-test split with stratified sampling. SVM parameters were set to kernel=linear and C=1.0; Naive Bayes used alpha=1.0 (Laplace smoothing). Experimental results show that SVM achieved an accuracy of 88.3%, precision of 0.89, recall of 0.88, and F1-score of 0.88, while Naive Bayes obtained an accuracy of 82.1%, precision of 0.84, recall of 0.82, and F1-score of 0.83. SVM demonstrated superior performance across all four evaluation metrics, with the largest gap observed in the F1-score for the negative class (SVM: 0.71 vs. Naive Bayes: 0.56). These findings confirm that SVM is more robust against class imbalance in informal Indonesian-language review data.*

Keywords *- Sentiment Analysis; Google Play Store; Naive Bayes; Support Vector Machine; TF-IDF;*

Analisis Komparatif Naive Bayes dan Support Vector Machine untuk Klasifikasi Sentimen Ulasan Aplikasi Mobile Berbahasa Indonesia pada Google Play Store

ABSTRAK *– Penelitian ini melakukan evaluasi performa komparatif antara Multinomial Naive Bayes dan Support Vector Machine (SVM) dengan kernel linear dalam mengklasifikasikan sentimen ulasan aplikasi mobile berbahasa Indonesia yang dikumpulkan dari Google Play Store. Sebanyak 2.847 ulasan terhadap aplikasi dompet digital GoPay dikumpulkan melalui teknik web scraping menggunakan library google-play-scraper. Setelah melewati proses preprocessing yang mencakup case folding, cleansing, tokenisasi, penghapusan stopword, dan stemming menggunakan pustaka Sastrawi, dataset final yang dapat digunakan berjumlah 2.634 ulasan. Pelabelan sentimen dilakukan secara otomatis berdasarkan rating bintang: rating 4 dan 5 dikategorikan sebagai sentimen positif (1.841 ulasan, 69,9%), sedangkan rating 1 dan 2 dikategorikan sebagai sentimen negatif (793 ulasan, 30,1%). Ekstraksi fitur menggunakan TF-IDF dengan ukuran kosakata 8.432 term unik. Pelatihan model menggunakan pembagian data 80:20 dengan stratified sampling. Parameter SVM ditetapkan pada kernel=linear dan C=1,0; Naive Bayes menggunakan alpha=1,0 (Laplace smoothing). Hasil eksperimen menunjukkan bahwa SVM mencapai akurasi 88,3%, presisi 0,89, recall 0,88, dan F1-score 0,88, sementara Naive Bayes memperoleh akurasi 82,1%, presisi 0,84, recall 0,82, dan F1-score 0,83. SVM menunjukkan performa lebih unggul pada seluruh empat metrik evaluasi, dengan selisih terbesar pada F1-score kelas negatif (SVM: 0,81 vs. Naive Bayes: 0,71). Temuan ini mengkonfirmasi bahwa SVM lebih robust terhadap ketidakseimbangan kelas pada data ulasan berbahasa Indonesia yang bersifat informal.*

Kata Kunci *– Analisis Sentimen; Google Play Store; Naive Bayes; Support Vector Machine; TF-IDF;*

Jurnal Ilmiah Informatika dan Komputer (Informatech)
Volume 3, Number 2026, June 2026, page 20-25
DOI : 10.69533
Website : *https://ejournal.rizaniamedia.com/index.php/informatech*
E-ISSN: 3047-4752

1. INTRODUCTION

The growth of the mobile application ecosystem in Indonesia in the past decade has created unique and challenging conditions for software developers. Data from Statista [1] shows that the number of mobile app downloads in Indonesia exceeded 7.8 billion in 2023, placing Indonesia as one of the largest app markets in Southeast Asia. Each potential installation leaves a trail of user reviews on distribution platforms such as the Google Play Store, which cumulatively form a huge but often unutilized corpus of data.

This ever-increasing volume of reviews is a real challenge. Noei, Zhang, and Zou [3] note that popular apps on the Google Play Store can receive thousands of new reviews every day, making manual reading and categorization operationally unrealistic. A review containing critical complaints about security bugs could sink among hundreds of short comments of a generic nature if there is no effective automatic screening mechanism. This is a relevant entry point for artificial intelligence-based sentiment analysis.

Sentiment analysis is a branch of Natural Language Processing (NLP) that aims to detect and classify opinions in text into specific classes of sentiments, generally positive, negative, or neutral. The two most widely used machine learning algorithms in this task are Naive Bayes and Support Vector Machine (SVM), each with different computational and performance characteristics [5], [6]. Naive Bayes excels in training speed and simplicity of implementation, while SVM is known to be more accurate in the space of high-dimensional features such as TF-IDF representations of text.

Although the two algorithms have been widely compared in the literature, most previous studies have not provided sufficiently transparent information regarding model parameters, validation strategies, data labeling processes, and class distribution. This condition makes replication and comparison difficult across studies. This research is here to fill this gap by presenting a controlled and fully documented experiment on the GoPay application review dataset on the Google Play Store. The main contributions of this study include: (1) the application of standardized star-rating-based labeling, (2) complete documentation of model parameters and validation strategies, (3) comprehensive evaluation using the confusion matrix and its four derivative metrics, and (4) performance analysis on real class imbalance conditions.

2. LITERATURE REVIEW

Research on mobile app review sentiment analysis using classic machine learning approaches has evolved significantly since the mid-2010s. Sari et al. [2] compared Naive Bayes and SVM for sentiment analysis of PUBG Mobile reviews on the Google Play Store and found that SVM consistently resulted in higher classification accuracy, especially under conditions of unbalanced class distribution. Leandro and Fianty [4] evaluated both algorithms for social media applications and reported that SVM showed superior precision when the feature space was high-dimensional.

Ghaffar [7] investigated Naive Bayes in the review domain of digital wallet applications in Indonesia and reported an accuracy value ranging from 78-84%, with performance gaps mainly due to the presence of non-standard language and code-mixing in the review corpus. Aufar et al. [8] demonstrated that a well-structured preprocessing pipeline, including TF-IDF weighting, significantly improves classification results without relying on specific algorithm choices.

Recent developments in sentiment analysis of Indonesian texts cannot be separated from the emergence of transformer-based models. IndoBERT, developed by Koto et al. [18] and published through the IndoNLU Benchmark, is a BERT-based pre-trained language model that is specially trained on large-scale Indonesian text corpus. In some recent tests, IndoBERT was able to achieve an accuracy of above 92% on sentiment classification tasks, far surpassing the TF-IDF feature-based approach. Although IndoBERT's accuracy is higher, its much greater computational complexity makes the classic machine learning approach still relevant, especially in resource-constrained environments.

CNN and LSTM have also been applied in the text sentiment classification. Liu et al. [15] demonstrated that LSTM-based deep learning architectures are capable of capturing sequential dependencies in text that traditional bag-of-words models can't reach, albeit with the consequence of greater training data needs. RoBERTa, as an optimized variant of BERT, has also been tested in review texts and consistently performs better than standard BERT under limited data conditions [20]. The selection of Naive Bayes and SVM in this study is based on their relevance as a well-established baseline, high interpretability, and lower computational cost compared to deep learning models.

Scholkopf and Smola [12] established the theoretical foundation that SVM maximizes the

margins between class boundaries in high-dimensional space, a particularly advantageous trait for sparse TF-IDF representations. Thakur, Tiwari, and Agrawal [14] confirm that linear kernels remain competitive across a wide range of text domains. This theoretical and empirical foundation reinforces the relevance of the comparison of the two algorithms in the specific context of Indonesian-language application reviews.

3. RESEARCH METHODOLOGY

This study was conducted at the Department of Informatics Engineering, Faculty of Engineering, Sam Ratulangi University Manado. Research activities included literature review, data collection, model development, system testing, and report writing.

3.1 DATA COLLECTION AND DATASET DESCRIPTION

The data used are primary data collected directly from the Google Play Store via web scraping using the Python-based google-play-scraper library. The collection target was reviews of the GoPay application (ID: com.gojek.gopay), one of the largest digital wallet applications in Indonesia with more than 50 million downloads. Collection was conducted without a time range restriction, with priority given to the most recent reviews. Attributes collected include review text (content), star rating (score), and review date (at).

The scraping process yielded 2,847 raw reviews. An initial selection process removed irrelevant entries including empty reviews, duplicates, and reviews containing only symbolic characters, leaving 2,634 usable reviews. Sentiment labeling was carried out automatically based on star ratings using the following scheme: ratings of 4 and 5 were categorized as positive sentiment, while ratings of 1 and 2 were categorized as negative sentiment. Reviews with a rating of 3 were excluded from the dataset as they are considered ambiguous and not representative of either class. The final dataset distribution is 1,841 positive reviews (69.9%) and 793 negative reviews (30.1%).

Table 1. GoPay Review Dataset Description

Sentiment Class	Number of Reviews	Percentage (%)
Positive (rating 4–5)	1,841	69.9%
Negative (rating 1–2)	793	30.1%
Total	2,634	100%

Examples of reviews from the dataset: positive reviews include "Aplikasi sangat mudah digunakan, transfer cepat dan aman" (rating 5), while negative reviews include "Saldo sudah dipotong tapi transaksi gagal terus, customer service tidak responsif" (rating 1).

3.2 TEXT MINING AND TEXT PREPROCESSING

Text mining is the process of transforming unstructured text data into numerical information that can be processed by computing machines. Given the high linguistic variation in Google Play Store reviews, which includes abbreviations, colloquial expressions, and code-mixing between Indonesian and English, text must be cleaned through preprocessing stages [8] consisting of: (1) Case Folding, converting all characters to lowercase; (2) Cleansing, removing non-alphabetic characters including URLs, numbers, and punctuation; (3) Tokenization, splitting sentences into individual word tokens; (4) Stopword Removal, eliminating common function words that carry no sentiment information; and (5) Stemming, reducing each token to its morphological root form using the Sastrawi library.

3.3 TF-IDF FEATURE EXTRACTION

After preprocessing, term weighting was performed using Term Frequency-Inverse Document Frequency (TF-IDF). The TfidfVectorizer module from Scikit-learn was used with the following parameters: min_df=2 (a term must appear in at least 2 documents), max_features=None, and sublinear_tf=True to reduce the dominance of very frequently occurring terms. The extraction resulted in a feature matrix with dimensions of 2,634 x 8,432, meaning there are 8,432 unique terms in the vocabulary after filtering. The TF-IDF weight is calculated using the formula:

$$Wij = TFij \times log(N / DFi) \quad (1)$$

3.4 CLASSIFICATION WITH NAIVE BAYES AND SVM

Model training was performed using two algorithms: Multinomial Naive Bayes and SVM with a linear kernel. Parameters set for SVM were kernel=linear and C=1.0 (default regularization parameter). Naive Bayes used alpha=1.0, which is the standard Laplace smoothing to avoid zero probabilities for terms not appearing in the training data. The validation strategy used was an 80:20 data split with stratified sampling, ensuring class distribution was maintained in both the training set (2,107 reviews) and the test set (527 reviews). Both models were trained separately using the same training data. Implementation used the MultinomialNB and SVC modules from Scikit-learn version 1.3.

4. RESULTS AND DISCUSSION

4.1 PREPROCESSING RESULTS AND FEATURE REPRESENTATION

Of the 2,847 raw reviews collected, the initial selection stage produced 2,634 clean reviews after removing 213 entries consisting of 87 duplicate reviews, 94 empty reviews or reviews containing only emojis, and 32 reviews with no meaningful text content. The final dataset distribution is 1,841 positive reviews (69.9%) and 793 negative reviews (30.1%), indicating an actual class imbalance condition with a ratio of approximately 2.3:1.

The preprocessing stage resulted in significant vocabulary reduction. Before stemming, the unique vocabulary numbered approximately 12,400 tokens. After applying Sastrawi stemming, the vocabulary shrunk to 8,432 unique terms. This reduction reflects the effectiveness of stemming in merging morphological variations such as "pembayaran" (payment), "bayar" (pay), and "dibayarkan" (paid) into a single root representation "bayar". Table 2 presents a summary of the results of each preprocessing stage.

Table 2. Preprocessing Stages and Their Impact on Features

Stage	Process	Impact on Features
Case Folding	Convert to lowercase	Reduces feature duplication
Cleansing	Remove numbers, symbols, URLs	Reduces non-linguistic noise
Tokenization	Split text into tokens	Foundation for vocabulary building
Stopword Removal	Remove non-informative words	Focuses features on sentiment words
Stemming (Sastrawi)	Reduce to root form	From ~12,400 to 8,432 unique terms
TF-IDF	Feature weighting	Matrix 2,634 × 8,432

4.2 CONFUSION MATRIX RESULTS

Testing was conducted on a test set of 527 reviews (421 positive and 106 negative). Tables 3 and 4 present the confusion matrices for each algorithm.

Table 3. Naive Bayes Confusion Matrix

	Predicted Positive	Predicted Negative
Actual Positive	TP = 374	FN = 47
Actual Negative	FP = 47	TN = 59

Table 4. Confusion Matrix (Linear Kernel)

	Predicted Positive	Predicted Negative
Actual Positive	TP = 388	FN = 33
Actual Negative	FP = 29	TN = 77

4.3 NAIVE BAYES PERFORMANCE COMPARISON

Based on the evaluation results in Table 5, the Multinomial Naive Bayes model achieved an overall accuracy of 82.1% (433 out of 527 correct predictions). This value indicates that the model is able to correctly classify most of the test data, but its performance is not uniform across each sentiment class. In terms of the weighted average, the model produces a weighted average of 0.84 for precision, 0.82 for recall, and 0.83 for F1-score.

For the positive class, Naive Bayes demonstrated relatively good performance. A precision value of 0.89 and recall of 0.89 indicate that the model is fairly consistent in identifying positive reviews. An F1-score of 0.88 also indicates a good balance between the model's ability to predict the positive class and recognize actual positive data.

In contrast, the model's performance on the negative class remains limited. The negative class only achieved a precision value of 0.56, recall of 0.56, and F1-score of 0.56. These results indicate that the model still struggles to distinguish negative reviews from other classes. This condition may occur due to the characteristics of Naive Bayes, which assumes independence between features, so sentence context is not always captured fully, particularly in reviews containing negations, informal language, or ambiguous sentence structures. Thus, Naive Bayes can be considered quite effective in classifying positive reviews, but suboptimal in detecting negative reviews. The performance disparity between the positive and negative classes indicates that this model still requires additional strategies, such as data balancing, normalization of non-standard words, or more representative feature selection to improve sensitivity to the negative class..

Table 5. Naive Bayes Evaluation Metrics per Class

Metric	Positive Class	Negative Class	Weighted Avg
Accuracy	82.1%	82.1%	82.1%
Precision	0.89	0.56	0.84
Recall	0.89	0.56	0.82
F1-Score	0.88	0.56	0.83

4.4 NAIVE BAYES PERFORMANCE COMPARISON

Based on the evaluation results in Table 6, the Support Vector Machine (SVM) model with a linear kernel achieved an overall accuracy of 88.3%. This value shows that the model was able to correctly classify 465 out of 527 test data points. In general, SVM's performance is considered good as it produced a weighted average of 0.89 for precision, 0.88 for recall, and 0.88 for F1-score.

When viewed by sentiment class, SVM's

performance on the positive class is higher than the negative class. The positive class obtained a precision value of 0.90, recall of 0.92, and F1-score of 0.90. These results indicate that the model has a strong ability to recognize positive reviews. Meanwhile, the negative class obtained a precision of 0.73, recall of 0.70, and F1-score of 0.71. These values indicate that the model's ability to detect negative reviews is still lower compared to the positive class.

Table 6. SVM (Linear Kernel) Evaluation Metrics per Class)

Metric	Positive Class	Negative Class	Weighted Avg
Accuracy	82.1%	82.1%	82.1%
Precision	0.89	0.56	0.84
Recall	0.89	0.56	0.82
F1-Score	0.88	0.56	0.83

4.5 Comparative Analysis and Performance Determinants

Based on Table 7, SVM with a linear kernel demonstrates superior performance compared to Naive Bayes across all major evaluation metrics. SVM achieved an accuracy of 88.3%, while Naive Bayes achieved an accuracy of 82.1%, a difference of 6.2 percentage points. This advantage is also evident in the weighted average precision, recall, and F1-score values: 0.89, 0.88, and 0.88 respectively for SVM, compared to 0.84, 0.82, and 0.83 for Naive Bayes.

The most notable difference is in the F1-score for the negative class. SVM achieved a value of 0.71, while Naive Bayes only achieved 0.56. A difference of 0.15 indicates that SVM is more effective at detecting negative reviews. This finding is practically important because negative reviews generally represent user complaints, service disruptions, or dissatisfaction with the application. If negative reviews are not classified properly, developers risk losing important information for improving service quality.

Table 7. Direct Comparison: Naive Bayes vs. SVM

Metric	Naive Bayes	SVM (Linear)	Difference	Best
Accuracy	82.1%	88.3%	+6.2%	SVM
Precision (wtd)	0.84	0.89	+0.05	SVM
Recall (wtd)	0.82	0.88	+0.06	SVM
F1-Score (wtd)	0.83	0.88	+0.05	SVM
F1 (negative class)	0.56	0.71	+0.15	SVM
Training Speed	Very Fast	Moderate	-	NB

The performance difference is influenced by the imbalanced data distribution and the linguistic characteristics of Indonesian-language Google Play Store reviews. The dataset is dominated by positive reviews at 69.9%, while negative reviews account for only 30.1%, making Naive Bayes more susceptible to bias toward the majority class as it relies on prior probabilities and token distributions. In addition, user reviews often contain abbreviations such as "gak", "ga", and "nggak", non-standard words, slang, and code-mixing. These linguistic variations can reduce the quality of feature representation, especially if normalization has not been able to standardize words with the same meaning. The impact is greater on Naive Bayes, whereas SVM is relatively more stable because it leverages geometric separation between classes even when features still contain language variations and textual noise.

5. Conclusion

This study successfully compared the performance of Naive Bayes and SVM in classifying the sentiment of Indonesian-language mobile application reviews on the Google Play Store through a fully documented pipeline, from web scraping of 2,847 GoPay reviews, Sastrawi-based preprocessing resulting in 2,634 clean reviews with 8,432 unique terms, star rating-based labeling yielding a distribution of 69.9% positive and 30.1% negative, to confusion matrix-based evaluation using an 80:20 data split with stratified sampling. Experimental results show that SVM with a linear kernel (C=1.0) consistently outperforms Naive Bayes (alpha=1.0) across all four evaluation metrics: accuracy 88.3% vs. 82.1%, weighted F1-score 0.88 vs. 0.83, with the largest gap in the negative class F1-score of 0.15 points (0.71 vs. 0.56). This most significant difference occurs because SVM is more robust to class imbalance and is better able to exploit the geometric structure of the sparse TF-IDF feature space. Naive Bayes remains relevant as a computationally efficient alternative for systems with resource limitations or requirements for very fast model training. The contribution of this study lies in providing a direct comparison of both algorithms with transparent and verified experimental documentation, including model parameter specifications, validation strategies, labeling processes, and complete confusion matrix data. Future research is recommended to explore transformer-based architectures such as IndoBERT for further comparison, given its ability to capture more complex semantic context. Expansion toward Aspect-Based Sentiment Analysis (ABSA) also has the potential to provide more granular insights for application developers.

BIBLIOGRAPHY

[1] Statista, "Number of mobile app downloads worldwide from 2016 to 2023," Statista Research Department, 2024. [Online]. Available: https://www.statista.com/statistics/271644/worldwide-free-and-paid-mobile-app-store-downloads/

[2] [P. R. Sari et al., "Comparison of Naive Bayes and SVM Algorithms for Sentiment Analysis of PUBG Mobile on Google Play Store," Sistemasi: Jurnal Sistem Informasi. [Online]. Available: http://sistemasi.ftik.unisi.ac.id

[3] [E. Noei, F. Zhang, and Y. Zou, "Too Many User-Reviews! What Should App Developers Look at First?," IEEE Transactions on Software Engineering, vol. 47, no. 2, pp. 367-378, Feb. 2021, doi: 10.1109/TSE.2019.2893171.

[4] J. O. Leandro and M. I. Fianty, "Evaluation of Sentiment Analysis Methods for Social Media Applications: A Comparison of SVM and Naive Bayes," International Journal on Informatics Visualization. [Online]. Available: www.joiv.org/index.php/joiv

[5] W. Lu, Y. Zhang, W. Wen, H. Yan, and C. Li, Eds., Cyber Security, vol. 1506. Singapore: Springer Nature Singapore, 2022, doi: 10.1007/978-981-16-9229-1.

[6] M. R. L. Cahya and E. Y. Hidayat, "Sentiment Analysis and Emotional Reviews of Hospital Services Using Naive Bayes and SVM," Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi, vol. 11, no. 1, pp. 121-129, Feb. 2026, doi: 10.25139/inform.v11i1.11257.

[7] S. A. Ghaffar, "Comparative Sentiment Analysis of Digital Wallet Applications in Indonesia Using Naive Bayes," IJIIS: International Journal of Informatics and Information Systems, vol. 8, no. 2, pp. 55-66, Mar. 2025, doi: 10.47738/ijiis.v8i2.251.

[8] A. F. Aufar, M. A. Rosid, A. Eviyanti, and I. R. I. Astutik, "Optimizing Text Preprocessing for Accurate Sentiment Analysis on E-Wallet Reviews," JICTE, vol. 7, no. 2, pp. 42-50, Oct. 2023, doi: 10.21070/jicte.v7i2.1650.

[9] B. Gunawan et al., "Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes," JEPIN, vol. 4, no. 2, pp. 17-29, 2018.

[10] M. Das, S. Kamalanathan, and P. Alphonse, "A Comparative Study on TF-IDF Feature Weighting Method and its Analysis using Unstructured Dataset," IJEAST, vol. 4, no. 11, 2020.

[11] D. A. Fatah et al., "Sentiment Analysis of Public Opinion Towards Tourism in Bangkalan Regency Using Naive Bayes Method," in E3S Web of Conferences, EDP Sciences, Mar. 2024, doi: 10.1051/e3sconf/202449901016.

[12] B. Scholkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, MA: MIT Press, 2002.

[13] A. M. Fajria, A. Faqih, and G. Dwilestari, "The Impact of Principal Component Analysis on Sentiment Classification Performance Using SVM," Journal of Artificial Intelligence and Engineering Applications, 2025. [Online]. Available: https://ioinformatic.org/

[14] S. Thakur, V. K. Tiwari, and J. Agrawal, "Performance Analysis of Linear Kernel SVM Models on Real-World Datasets," Int. J. Advanced Networking and Applications, 2025.

[15] J. Liu, Z. Liu, Q. Li, W. Kong, and X. Li, "Multi-Domain Controversial Text Detection Based on a Machine Learning and Deep Learning Stacked Ensemble," Mathematics, vol. 13, no. 9, May 2025, doi: 10.3390/math13091529.

[16] N. F. Hidayah, K. R. P. Kartika, and S. N. Budiman, "Penerapan Metode Naive Bayes dalam Analisis Sentimen Aplikasi Sentuh Tanahku pada Google Play," 2022.

[17] P. M. N. Dharmapatni and N. L. P. Merawati, "Penerapan Algoritma Support Vector Machine dalam Sentimen Analisis Terkait Kenaikan Tarif BPJS Kesehatan," Jurnal Bumigora Information Technology (BITe), vol. 2, no. 2, pp. 105-112, Sep. 2020, doi: 10.30812/bite.v2i2.904.

[18] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding," in Proc. 1st AACL-IJCNLP, Dec. 2020, pp. 843-857. [Online]. Available: https://arxiv.org/abs/2009.05387

[19] A. D. Fitriyanto and P. Purwanto, "Analisis Sentimen Ulasan DANA dari Play Store dengan Metode SVM, Logistic Regression, Naive Bayes dan KNN," Building of Informatics, Technology and Science (BITS), vol. 7, no. 3, pp. 1887-1899, Dec. 2025, doi: 10.47065/bits.v7i3.8769.

[20] M. M. Rahman, A. I. Shiplu, Y. Watanobe, and M. A. Alam, "RoBERTa-BiLSTM: A Context-Aware Hybrid Model for Sentiment Analysis," arXiv preprint arXiv:2406.00367, 2024. [Online]. Available: https://arxiv.org/abs/2406.00367 M. M. Rahman, A. I. Shiplu, Y. Watanobe, and M. A. Alam, "RoBERTa-BiLSTM: A Context-Aware Hybrid Model for Sentiment Analysis," arXiv preprint arXiv:2406.00367, 2024. [Online]. Available: https://arxiv.org/abs/2406.00367