

Evaluation of Machine Learning Algorithms in Sentiment Analysis of the Satu Sehat Application

Marwan Suhendra¹, Badariatul Lailiah², Yanto³, Lady Agustin Fitriana^{4*}

^{1,3}Program Studi Informatika,

^{2,4}Program Studi Sistem Informasi

Universitas Bina Sarana Informatika

Lady.lag@gmail.com*

Abstract

This study aims to analyze and compare the performance of three sentiment classification algorithms—Support Vector Machine (SVM), Naïve Bayes (NB), and K-Nearest Neighbor (K-NN)—in classifying user reviews of the Satu Sehat application. The data preprocessing stage involves several steps, including text cleaning through normalization, removal of punctuation, numbers, and irrelevant characters, as well as the elimination of stopwords. Subsequently, stemming is performed to reduce words to their root forms. Feature extraction is conducted using the CountVectorizer method with a bag-of-words approach, which converts textual data into numerical representations. The dataset is then divided into training and testing subsets using an 80:20 train-test split ratio. Model performance is evaluated through a confusion matrix, producing key evaluation metrics such as accuracy, precision, recall, and F1-score. Based on the results of testing 9,192 user reviews, the SVM algorithm with a linear kernel demonstrated the best overall performance compared to NB and K-NN, as indicated by the highest accuracy score. These findings suggest that SVM is more effective in handling high-dimensional textual features, making it a highly suitable algorithm for sentiment analysis of digital health application reviews, particularly those related to Satu Sehat.

Keywords: *Sentiment Analysis, Satu Sehat, Support Vector Machine (SVM), Naïve Bayes (NB), K-Nearest Neighbor (K-NN)*

1. Introduction

The rapid advancement of digital technology has brought significant transformation to the healthcare sector. One notable manifestation of this transformation is the development of Satu Sehat, an application initiated by the Ministry of Health of the Republic of Indonesia. The platform is designed to integrate medical records, vaccination data, and public health services into a single, unified digital ecosystem. As the number of users continues to grow, public opinions shared through digital platforms have emerged as a valuable source of information for assessing the quality of services and understanding the overall user experience [1].

Sentiment analysis is a subfield of Natural Language Processing (NLP) that focuses on classifying user opinions into positive, negative, or neutral sentiments [2]. This method has been widely applied across various domains, including e-commerce, public transportation, government services, and, more recently, within the digital healthcare sector. In sentiment classification tasks, the selection of an appropriate machine learning algorithm plays a crucial role in enhancing the accuracy and reliability of the analytical outcomes [3].

Several previous studies, including one conducted by Aina Damayunita et al., examined the performance of three classification algorithms Naïve Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) to identify patients with heart disease based on various factors such as Body Mass Index (BMI) and skin cancer status. The results indicated that SVM achieved the highest accuracy at 92%, while Naïve Bayes recorded the lowest accuracy at 88%[4]. A subsequent study conducted by Francis Matheos Sarilome compared the performance of Naïve Bayes and Support Vector Machine (SVM) algorithms in performing sentiment analysis on the Satu Sehat application using Twitter data comprising more than 1,046 samples. The results demonstrated that SVM achieved the highest accuracy of 87.95%, surpassing Naïve Bayes, which attained an accuracy of 81.56%. These findings indicate that SVM is more efficient and accurate in classifying public sentiment related to information and educational content concerning the Satu Sehat application [5].

Finally, a study conducted by Aulia Putri et al. compared the performance of K-Nearest Neighbor (K-NN), Naïve Bayes, and Support Vector Machine (SVM) algorithms in predicting the graduation outcomes of final-year students, using data collected from 379 respondents. The findings revealed that K-NN achieved the best performance, with an accuracy of 87.8%, precision of 87.8%, and recall of 84%. This algorithm was considered the most effective in predicting postgraduate student graduation, as it was able to account for various campus-related factors that influence timely completion of studies [6].

Based on the aforementioned studies, this research aims to compare and evaluate the performance of the Naïve Bayes, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) algorithms in conducting sentiment analysis on user reviews of the Satu Sehat application. The findings of this study are expected to contribute to generating a more accurate understanding of public opinion, thereby supporting efforts to improve the quality of healthcare services in Indonesia.

2. Theoretical Foundation

2.1 Analisis Sentimen

Sentiment analysis is an automated method used to extract and interpret opinions from textual data. This technique collects online statements related to a brand or product and subsequently analyzes them to determine whether the expressed sentiment is positive, negative, or neutral. Through this analysis, it becomes possible to observe changes in consumer attitudes over time, providing valuable insights into how public perception evolves. Among the algorithms frequently applied in this process is Naïve Bayes, which serves as one of the fundamental approaches for text-based sentiment classification [7].

2.2 Naïve Bayes

Naïve Bayes is a probabilistic classification algorithm that applies Bayes' theorem under the assumption that each feature is independent of the others. This method is widely recognized for its efficiency and speed in processing textual data, as well as its strong performance in sentiment classification tasks, even when dealing with high-dimensional datasets. Numerous studies have demonstrated that Naïve Bayes can achieve satisfactory accuracy across various sentiment analysis applications, thereby supporting the success of marketing strategies and contributing to the enhancement of product reputation through accurate interpretation of public opinion [8].

2.3 K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) is a supervised machine learning method used for both classification and regression tasks, which operates based on the proximity or distance between data points. The algorithm works by identifying the K nearest neighbors of a given test instance and determining its class according to the majority label among those neighbors. KNN is categorized as an instance-based and lazy learning algorithm because it does not construct an explicit model from the training data; instead, it relies on storing the training instances and making predictions only when new data are introduced [9].

2.4 Support Vector Machine (SVM)

Support Vector Machine (SVM) is an algorithm that focuses on finding the optimal hyperplane to separate classes within a high-dimensional feature space. Several studies have shown that SVM generally achieves higher accuracy levels compared to other methods in sentiment classification tasks, owing to its ability to handle complex and non-linear data distributions effectively [10].

3. Research Method

This study employs an experimental comparative approach aimed at evaluating and comparing the performance of Naïve Bayes, K-Nearest Neighbor (KNN), and Support Vector Machine (SVM) algorithms in conducting sentiment analysis on user reviews of the Satu Sehat application. The overall research process is visualized through a flowchart, as illustrated in the attached figure.

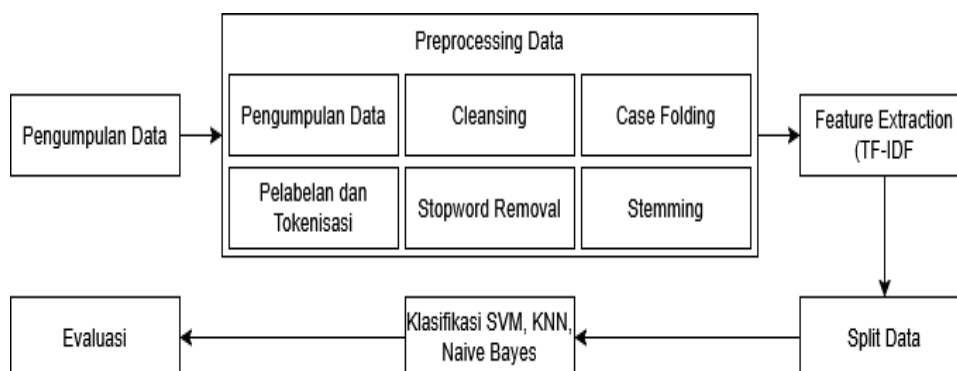


Figure 1. Research Stages

3.1 Data Collection

The data collection process was conducted using a scraper built with the `google-play-scraper` library to automatically extract user reviews of the Satu Sehat application from the Google Play Store. The extraction was performed based on the application ID (`com.telkom.tracencare`), with filters set to the Indonesian language and Indonesia region, and the reviews were sorted by most recent. A total of 10,000 user reviews were successfully retrieved, converted into a Pandas DataFrame, and stored in CSV format. The dataset was then displayed and analyzed directly using Google Colab, facilitating efficient data exploration and preprocessing.

```
from google_play_scraper import reviews, Sort

app_id = 'com.telkom.tracencare'

def get_reviews(app_id, lang='id', count=10000, sort=Sort.
try:
    result, continuation_token = reviews(
        app_id,
        lang=lang,
        country='id',
        sort=sort,
        count=count,
        filter_score_with=filter_score_with,
        filter_device_with=filter_device_with,
        continuation_token=continuation_token
    )
```

Figure 2. Scraping Process

3.2 Preprocessing

The data preprocessing process consists of several main stages:

3.2.1 Cleansing

Data cleansing, also known as data cleaning, is the process of detecting and correcting (or removing) incorrect, inaccurate, incomplete, irrelevant, or corrupted data within a dataset. The primary objective of this stage is to improve data quality so that it is ready for further analysis or processing. This step involves the removal of irrelevant elements such as punctuation marks, numerical values, and single characters, as well as the elimination of extra spaces, resulting in cleaner and more structured textual data that can be more effectively analyzed in subsequent stages.

3.2.2 Case Folding

The case folding stage refers to the process of converting all letters in a text into either lowercase or uppercase format. In this study, all textual data were converted to lowercase to ensure consistency in writing format and to prevent semantic discrepancies that might arise due to variations in letter capitalization.

3.2.3 Word Normalization

Word normalization aims to standardize text representation by replacing non-standard words or word variants (such as slang or abbreviations) with their standardized forms. This step is essential for reducing lexical variation and enhancing data consistency, thereby improving the overall quality of the text for subsequent analysis.

3.2.4 Data Labeling and Tokenization

Data labeling in sentiment analysis is the process of assigning labels to review or textual data to classify them into specific sentiment categories, such as positive, negative, or neutral. The main objective of this stage is to simplify the analytical process and enable a deeper understanding of the perceptions or opinions conveyed in the text. Tokenization refers to the process of breaking down text into smaller linguistic units. In this study, each token represents an individual word extracted from user reviews. The tokenization process was implemented using the `word_tokenize` function from the Natural Language Toolkit (NLTK) library in Python, which is a comprehensive toolkit for natural language processing (NLP) tasks. To integrate this function into the preprocessing stage, a wrapper function named `word_tokenize_wrapper(text)` was defined, which takes a text input and produces a list of tokens as its output.

3.2.5 Stopword Removal

Stopword removal is a technique used to eliminate common words such as “and”, “or”, and “is”, as these words contribute very little or in some cases, nothing at all to the sentiment analysis process. By removing such words, the analysis can focus more effectively on relevant and meaningful terms that carry greater semantic weight. In this study, stopwords in both Indonesian and English were removed using the default stopword lists provided by NLTK, which were then combined with an additional manually curated list stored in a file named `stopwords.csv`. The integration of these two lists aimed to ensure comprehensive coverage of stopwords relevant to the specific context of user review data. This process was implemented through the `stopword_removal(Review)` function, which filters the token list and removes any token present in the stopword list.

3.3 Stemming

Stemming is the process of reducing words to their root or base form. In this study, stemming was performed using the Sastrawi library, which is specifically designed for processing Indonesian-language text. The stemming(Review) function applies the stemming procedure to each token in the dataset and returns the output as a concatenation of root words, producing cleaner and more standardized text for subsequent analysis.

3.4 Feature Extraction (TF-IDF)

This stage is conducted to convert textual data into a numerical representation that can be processed by machine learning models, followed by calculating the Term Frequency–Inverse Document Frequency (TF-IDF) weight for each word in the review. The TF-IDF technique assigns higher weights to words that occur frequently within a particular document but rarely across the entire corpus, thereby highlighting terms that are most relevant for distinguishing one review from another.

3.5 Split Data

The dataset was divided into two subsets: 80% for training data and 20% for testing data. This division aims to evaluate the model's ability to generalize to unseen data. The data splitting process was implemented using the `train_test_split` function from the Scikit-learn library.

3.6 Classification

This stage involves the implementation of three algorithms: Support Vector Machine (SVM), K-Nearest Neighbor (K-NN), and Naïve Bayes. The SVM algorithm operates by identifying the optimal hyperplane that separates data classes, while K-NN performs classification based on the proximity or similarity between data points, and Naïve Bayes applies a probabilistic approach grounded in Bayes' theorem. These three methods were evaluated to determine the most effective algorithm for classifying user review sentiments.

3.6.1 Support Vector Machine (SVM)

One of the algorithms used for classification is the Support Vector Machine (SVM). This method seeks to identify the optimal separating function (hyperplane) that best distinguishes between two data categories. SVM is widely recognized for its robustness against overfitting, particularly when operating in high-dimensional feature spaces. The algorithm is based on the principle of Structural Risk Minimization (SRM), which aims to reduce the probability of classification errors. Originally developed as a linear classifier, SVM was later extended to handle non-linear cases through the implementation of various kernel functions [14].

3.6.2 K-Nearest Neighbor (KNN)

The K-Nearest Neighbor (K-NN) algorithm is a widely used classification method that determines the class of a data point based on its k nearest neighbors. The process of identifying these neighbors is carried out by calculating the distance between data instances. The class that appears most frequently among the nearest neighbors is then assigned as the predicted label for the test data [15].

3.6.3 Naïve Bayes

Naïve Bayes is a probabilistic classification algorithm known for its simplicity and effectiveness. This approach operates by calculating the probability of a class based on the frequency and combination of feature values within a dataset. Naïve Bayes assumes that each attribute is independent of the others, meaning that there are no interdependencies among features during the probability computation process [16].

3.7 Evaluasi

Model evaluation aims to measure the ability of an algorithm to accurately classify sentiments. This stage employs several performance metrics, including accuracy, precision, recall, and F1-score, which are calculated using the `classification_report` and `accuracy_score` functions from the Scikit-learn library. Additionally, a confusion matrix is utilized to visualize the distribution of prediction errors across different sentiment classes. This evaluation process provides a comprehensive overview of each model's performance in recognizing textual patterns and serves as a common approach in sentiment analysis research to determine the most optimal classification algorithm.

4. Result and Discussion

4.1 Data Collection

The scraped dataset contains the user name, rating, upload time, and review text of the Satu Sehat application, filtered by language (Indonesian) and region (Indonesia), and sorted by the most recent reviews. The collected data were then converted into a Pandas DataFrame, saved in CSV format, and displayed in Google Colab for verification and subsequent analysis.

	Review ID	Username	Rating	Review Text	Date
0	6781ff89-1c9d-4325-8b45-56379710bbfb	Pengguna Google	1	gak bisa login pake akun doang dih	2025-10-12 02:47:31
1	41568cd5-53fc-4dcc-9df6-c974ce9b6602	Pengguna Google	1	aplikasi negara kah ini? klo iya knp gangguan ...	2025-10-11 23:43:08
2	3e9102a4-ede0-4fae-a55a-26cf66591195	Pengguna Google	1	Aku belum cek kesehatan gratis, tapi kenapa ma...	2025-10-11 10:46:13
3	583c8666-ccf8-418e-9370-8adc571c06d0	Pengguna Google	1	ini apk apaan otp dikirim benar ga sesuai teru...	2025-10-11 00:31:25

Figure 3. Scraping Results of the Satu Sehat Application

4.2 Preprocessing Data

This process stage includes several preprocessing steps: data cleaning to remove unnecessary characters, case folding to convert all letters to lowercase, normalization to replace non-standard words with their standard forms, tokenization to split sentences into individual words, stopword removal to eliminate common words with little analytical value, and stemming to reduce inflected words to their root forms using the Sastrawi stemmer for the Indonesian language

4.2.1 Cleansing

At the cleansing stage, the review texts were cleaned by removing URLs, HTML tags, mentions, emojis, symbols, and numeric characters using regular expressions. This process resulted in cleaner textual data, making it more suitable for subsequent sentiment analysis.

Review Text	cleaning
gak bisa login pake akun doang dih	gak bisa login pake akun doang dih
aplikasi negara kah ini? klo iya knp gangguan ...	aplikasi negara kah ini klo iya knp gangguan m...
Aku belum cek kesehatan gratis, tapi kenapa ma...	Aku belum cek kesehatan gratis tapi kenapa mal...
ini apk apaan otp dikirim benar ga sesuai teru...	ini apk apaan otp dikirim benar ga sesuai teru...

Figure 4. Results of the Cleansing Process

4.2.2 Case Folding

At the case folding stage, all review texts were converted to lowercase to ensure consistency in text formatting and to avoid semantic discrepancies caused by capitalization differences. This process standardizes the textual data, making it more consistent and ready for the subsequent stages of sentiment analysis..

cleaning	case_folding
gak bisa login pake akun doang dih	gak bisa login pake akun doang dih
aplikasi negara kah ini klo iya knp gangguan m...	aplikasi negara kah ini klo iya knp gangguan m...
Aku belum cek kesehatan gratis tapi kenapa mal...	aku belum cek kesehatan gratis tapi kenapa mal...
ini apk apaan otp dikirim benar ga sesuai teru...	ini apk apaan otp dikirim benar ga sesuai teru...

Figure 5. Results of the Case Folding Process

4.2.3 Word Normalization

At the word normalization stage, non-standard words in the review texts were replaced with their standardized forms based on a predefined dictionary. This process ensures that the words within the text are more consistent and linguistically appropriate, thereby improving the overall data quality before proceeding to the sentiment analysis stage.

case_folding	normalisasi
gak bisa login pake akun doang dih	gak bisa login pake akun doang dih
aplikasi negara kah ini klo iya knp gangguan m...	aplikasi negara kah ini klo iya knp gangguan m...
aku belum cek kesehatan gratis tapi kenapa mal...	aku belum cek kesehatan gratis tapi kenapa mal...
ini apk apaan otp dikirim benar ga sesuai teru...	ini apk apaan otp dikirim benar ga sesuai teru...

Figure 6. Results of the Word Normalization Process

4.2.4 Tokenization

At the tokenization stage, the review texts were segmented into individual word units (tokens) using space-based separation. This process aims to break sentences into discrete words, thereby facilitating further analysis such as feature extraction and sentiment modeling.

normalisasi	tokenize
gak bisa login pake akun doang dih	[gak, bisa, login, pake, akun, doang, dih]
aplikasi negara kah ini klo iya knp gangguan m...	[aplikasi, negara, kah, ini, klo, iya, knp, ga...]
aku belum cek kesehatan gratis tapi kenapa mal...	[aku, belum, cek, kesehatan, gratis, tapi, ken...]
ini apk apaan otp dikirim benar ga sesuai teru...	[ini, apk, apaan, otp, dikirim, benar, ga, ses...]

Figure 7. Results of the Tokenization Process

4.2.5 Stopword Removal

At this stage, Indonesian stopwords were obtained from the NLTK library to identify common words that do not contribute significantly to sentiment analysis, such as “dan” (and), “yang” (which/that), or “di” (in/at). This process aims to prepare a list of words to be removed in the subsequent step, ensuring that only semantically meaningful and relevant terms are retained for modeling and analysis.

tokenize	stopword removal
[gak, bisa, login, pake, akun, doang, dih]	[gak, login, pake, akun, doang, dih]
[aplikasi, negara, kah, ini, klo, iya, knp, ga...]	[aplikasi, negara, kah, klo, iya, knp, ganggua...]
[aku, belum, cek, kesehatan, gratis, tapi, ken...]	[cek, kesehatan, gratis, hasil, pemeriksaannya...]
[ini, apk, apaan, otp, dikirim, benar, ga, ses...]	[apk, otp, dikirim, ga, sesuai, masuk, nya, gm...]

Figure 8. List of Indonesian Stopwords Extracted from NLTK

4.2.6 Stemming

At this stage, the Sastrawi library was installed, and the stemming modules from both Sastrawi and NLTK were imported to facilitate the stemming process. Stemming is used to convert words into their root or base form, thereby standardizing variations of words that share

similar meanings. This preparation step ensures that Indonesian language processing operates more effectively and optimally during sentiment analysis.

stopword removal	stemming_data
[gak, login, pake, akun, doang, dih]	gak login pake akun doang dih
[aplikasi, negara, kah, klo, iya, knp, ganggua...]	aplikasi negara kah klo iya knp ganggu mulu si...
[cek, kesehatan, gratis, hasil, pemeriksaannya...]	cek sehat gratis hasil periksa puskesmas tanju...
[apk, otp, dikirim, ga, sesuai, masuk, nya, gm...]	apk otp kirim ga sesuai masuk nya gmna gejala

Figure 9. Results of the Stemming Process

4.3 Labelling Data

At the data labeling stage, sentiment labeling was carried out using a lexicon-based method with the InSet (Indonesian Sentiment Lexicon). Each Satu Sehat application review was analyzed based on the number of positive and negative words identified after the preprocessing phase. The sentiment label for each review was determined by comparing these counts: reviews containing more positive words were labeled as "Positive", those with more negative words as "Negative", and reviews with an equal number of positive and negative words as "Neutral."

Rating	stemming_data	Sentiment	Score
0	gak login pake akun doang dih	Netral	0
1	aplikasi negara kah klo iya knp ganggu mulu si...	Negatif	-2
2	cek sehat gratis hasil periksa puskesmas tanju...	Netral	0
3	apk otp kirim ga sesuai masuk nya gmna gejala	Positif	1

Figure 10. Results of the Labeling Process

4.4 Split Data

The data splitting process was carried out using the train_test_split function with a ratio of 80% training data and 20% testing data. From the total dataset, 7,353 entries were allocated for training and 1,839 entries for testing. This division was intended to ensure that the model could learn effectively from the training data while being evaluated objectively using the testing data. In addition, the stratify parameter was applied to maintain a balanced distribution of sentiment classes positive, negative, and neutral across both subsets.

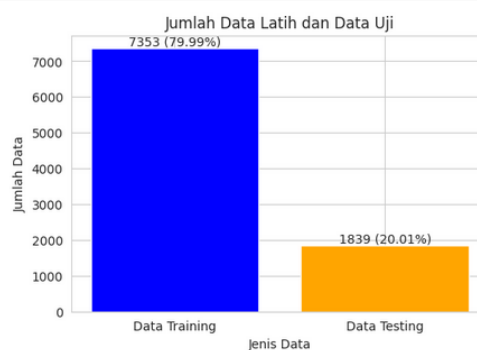


Figure 11. Split Data

4.5 Algoritma

4.5.1 Support Vector Machine (SVM)

Based on the classification report, the SVM model demonstrated excellent performance with an accuracy of 91.8%. The Negative class achieved the best results, with a precision of 0.964, recall of 0.958, and F1-score of 0.961, indicating a strong capability in recognizing negative sentiments. The Neutral and Positive classes also performed well, each obtaining an F1-score of 0.863. Furthermore, the macro average F1-score of 0.896 suggests that the model was able to classify the three sentiment categories in a balanced manner, while the weighted average F1-score of 0.919 reflects the model's consistent performance across the dataset's distribution.

SVM Confusion Matrix

Actual	Negatif	1000	40	4
	Netral	33	447	41
	Positif	4	28	242
		Negatif	Netral	Positif
		Predicted		

Figure 12. Confusion Matrix SVM.

4.5.2 K-Nearest Neighbor (KNN)

The classification report for the K-Nearest Neighbor (KNN) model indicates an accuracy of 66.7%, reflecting a moderate performance that remains inferior to the SVM model. The Negative class achieved the best results with an F1-score of 0.774, while the Neutral class exhibited high recall but low precision, suggesting that misclassifications occurred relatively frequently. The Positive class recorded the lowest performance, with an F1-score of 0.365. Overall, the macro average F1-score of 0.577 and the weighted average F1-score of 0.662 indicate that the model lacks stability and tends to be biased toward the dominant class.

KNN Confusion Matrix

Actual	Negatif	755	282	7
	Netral	105	406	10
	Positif	47	162	65
		Negatif	Netral	Positif
		Predicted		

Figure 13. Confusion Matrix KNN

4.5.3 Naïve Bayes

Based on the classification report, the Naïve Bayes model achieved an accuracy of 64.9%, indicating a moderate level of performance. The Negative class produced the best results with a precision of 0.646, recall of 0.970, and an F1-score of 0.776, reflecting strong capability in detecting negative sentiments but also a tendency toward misclassification. In contrast, the Neutral and Positive classes performed relatively poorly, with the Neutral class recording the lowest F1-score of 0.256.

The macro average F1-score of 0.503 and the weighted average F1-score of 0.584 suggest that the model's performance is imbalanced and biased toward the Negative class.

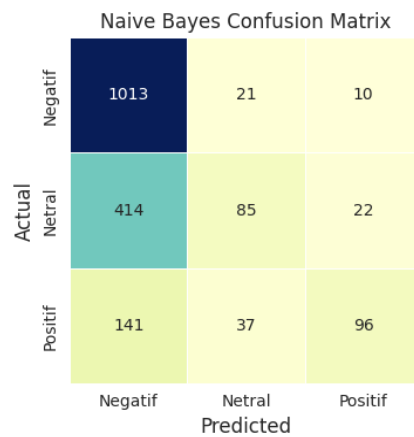


Figure 14. Confusion Matrix Naïve Bayes

4.6 Evaluation

The performance evaluation of the Naïve Bayes, K-Nearest Neighbor (K-NN), and Support Vector Machine (SVM) models was conducted using a confusion matrix to derive key performance metrics, including accuracy, precision, recall, and F1-score. The evaluation process employed the hold-out validation method, with the dataset split into 80% for training and 20% for testing. The visualization of classification results, as illustrated in the corresponding figure, presents the confusion matrix for each model, where the actual values represent the true sentiment labels and the predicted values indicate the model's classification outputs. Naïve Bayes, K-Nearest Neighbor (K-NN), dan Support Vector Machine (SVM) dilakukan menggunakan confusion matrix untuk memperoleh metrik accuracy, precision, recall, dan F1-score. Proses evaluasi menerapkan metode hold-out validation dengan pembagian data 80% pelatihan dan 20% pengujian. Visualisasi hasil klasifikasi pada Gambar menampilkan confusion matrix masing-masing model, di mana nilai actual menunjukkan label asli dan predicted menunjukkan hasil prediksi model.

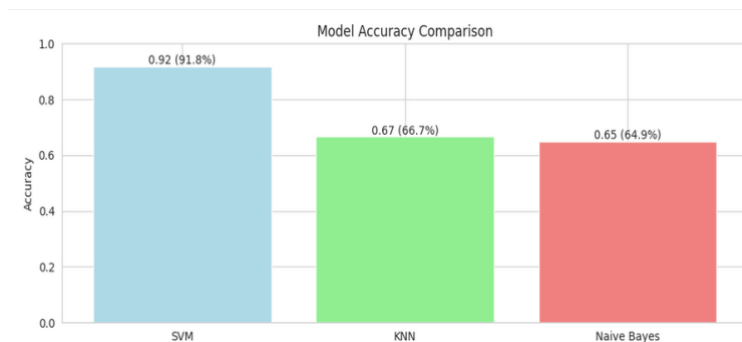


Figure 15. Performance Results of SVM, KNN, and NBC

The table 1 shows that the Support Vector Machine (SVM) model outperformed both K-Nearest Neighbor (KNN) and Naïve Bayes Classifier (NBC) in all evaluation metrics. SVM achieved the highest accuracy (0.918) and balanced performance across macro and weighted averages, whereas KNN and NBC produced moderate results with lower F1-scores.

Table 1 Comparison of Model Performance

Metrik	SVM	KNN	NBC
Accuracy	0.918	0.667	0.649
Macro Precision	0.892	0.701	0.663
Macro Recall	0.900	0.580	0.495
Macro F1	0.896	0.577	0.503
Weighted F1	0.919	0.662	0.584

5. Conclusion

This study aimed to compare the performance of three sentiment classification algorithms Support Vector Machine (SVM), Naïve Bayes (NB), and K-Nearest Neighbor (K-NN) in analyzing user reviews of the Satu Sehat application. From a total of 9,192 reviews, the SVM model with a linear kernel achieved the best performance, attaining an accuracy of 91.8%, followed by K-NN (66.7%) and Naïve Bayes (64.9%). The superiority of SVM lies in its ability to construct an optimal hyperplane in high-dimensional feature spaces, enabling it to more effectively capture complex sentiment patterns and contextual nuances. In contrast, the NB and K-NN algorithms exhibited limitations

in handling semantic variations and implicit expressions. These findings indicate that SVM is the most efficient algorithm for sentiment analysis of digital health application reviews, reinforcing its relevance in the advancement of Indonesian-language Natural Language Processing (NLP). For future research, it is recommended to employ a more diverse dataset to improve model generalization across various types of reviews and linguistic contexts. Additionally, sentiment labeling methods may be diversified such as through rating-based labeling, VADER, or other lexicon-based approaches to enable a more comprehensive comparative evaluation of classification results. The exploration of alternative algorithms such as Random Forest, XGBoost, or Deep Learning models (e.g., LSTM and BERT) is also encouraged to assess potential improvements in accuracy and the model's ability to capture deeper linguistic and contextual representations.

References

- [1] Y. M. Geasela, M. F. Isputrawan, F. S. Lee, and E. Christian, "SISTEM PENDAFTARAN TERINTEGRASI SATUSEHAT UNTUK AKSES LAYANAN KESEHATAN MENUJU INDONESIA SEHAT 2045," *J. Inform.*, vol. 8, no. 4, 2024.
- [2] A. Nurian, "ANALISIS SENTIMEN ULASAN PENGGUNA APLIKASI GOOGLE PLAY MENGGUNAKAN NAÏVE BAYES," *J. Inform. Dan Tek. Elektro Terap.*, vol. 11, no. 3s1, Sept. 2023, doi: 10.23960/jitet.v11i3s1.3348.
- [3] Lady Agustin Fitriana, M. Fahmi Julianto, R. Dahlia, M. Rifqi Firdaus, and A. Fazriansyah, "Analisis Ulasan Konsumen sebagai Data Non-Kuangan dalam Sistem Informasi Akuntansi," *Profitabilitas*, vol. 5, no. 1, pp. 64–74, July 2025, doi: 10.31294/profitabilitas.v5i1.8269.
- [4] Israt Jahan, Md Nakibul Islam, Md Mahadi Hasan, and Md Rafiuddin Siddiky, "Comparative analysis of machine learning algorithms for sentiment classification in social media text," *World J. Adv. Res. Rev.*, vol. 23, no. 3, pp. 2842–2852, Sept. 2024, doi: 10.30574/wjarr.2024.23.3.2983.
- [5] F. Matheos Sarimole and K. Kudrat, "Analisis Sentimen Terhadap Aplikasi Satu Sehat Pada Twitter Menggunakan Algoritma Naive Bayes Dan Support Vector Machine: Analisis Sentimen," *J. Sains Dan Teknol.*, vol. 5, no. 3, pp. 783–790, Feb. 2024, doi: 10.55338/saintek.v5i3.2702.
- [6] A. Putri *et al.*, "Komparasi Algoritma K-NN, Naive Bayes dan SVM untuk Prediksi Kelulusan Mahasiswa Tingkat Akhir: Comparison of K-NN, Naive Bayes and SVM Algorithms for Final-Year Student Graduation Prediction," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 1, pp. 20–26, May 2023, doi: 10.57152/malcom.v3i1.610.
- [7] A. Hendra and F. Fitriyani, "Analisis Sentimen Review Halodoc Menggunakan Naive Bayes Classifier," *JISKA J. Inform. Sunan Kalijaga*, vol. 6, no. 2, pp. 78–89, May 2021, doi: 10.14421/jiska.2021.6.2.78-89.
- [8] P. Nabila, F. Wajidi, and W. Firgiawan, "Analisis Algoritma K-Nearest Neighbor dan Naive Bayes pada Kebijakan Perkuliahan Online dengan Multi Bahasa," *Techno.Com*, vol. 24, no. 2, pp. 524–542, May 2025, doi: 10.62411/tc.v24i2.12656.
- [9] F. M. N. Akbar, "Metode KNN (K-Nearest Neighbor) untuk Menentukan Kualitas Air," *J. Tekno Kompak*, vol. 18, no. 1, p. 28, Feb. 2024, doi: 10.33365/jtk.v18i1.3241.
- [10] M. H. Arfian *et al.*, "Analisis Sentimen Pada Media Sosial Menggunakan Metode Support Vector Machine," *J. Ilmu Tek. Dan Komput.*, vol. 9, no. 1, pp. 1–6, Jan. 2025.
- [11] Fauziah, M. A. Tiro, and Ruliana, "Comparison of k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM) Methods for Classification of Poverty Data in Papua," *ARRUS J. Math. Appl. Sci.*, vol. 2, no. 2, pp. 83–91, Mar. 2022, doi: 10.35877/mathscience741.
- [12] S. K. P. Loka and A. Marsal, "Perbandingan Algoritma K-Nearest Neighbor dan Naive Bayes Classifier untuk Klasifikasi Status Gizi Pada Balita: Comparison Algorithm of K-Nearest Neighbor and Naive Bayes Classifier for Classifying Nutritional Status in Toddlers," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 1, pp. 8–14, May 2023, doi: 10.57152/malcom.v3i1.474.
- [13] M. H. Asnawi, I. Firmansyah, R. Novian, and R. S. Pontoh, "Perbandingan Algoritma Naive Bayes, K-NN, dan SVM dalam Pengklasifikasian Sentimen Media Sosial," 2021.
- [14] P. D. Silitonga and R. Damanik, "Perbandingan Algoritma k-Nearest Neighbors (k-NN) dan Support Vector Machines (SVM) untuk Klasifikasi Pengenalan Citra Wajah," *J. ICT Inf. Commun. Technol.*, vol. 20, no. 1, pp. 186–191, July 2021, doi: 10.36054/jict-ikmi.v20i1.354.
- [15] A. Tangkelayuk, "The Klasifikasi Kualitas Air Menggunakan Metode KNN, Naive Bayes, dan Decision Tree," *JATISI J. Tek. Inform. Dan Sist. Inf.*, vol. 9, no. 2, pp. 1109–1119, June 2022, doi: 10.35957/jatisi.v9i2.2048.
- [16] J. Wibisono, A. Suharsono, and S. H. Wijoyo, "PERBANDINGAN KINERJA METODE NAIVE BAYES DAN K- NEAREST NEIGHBOR UNTUK KLASIFIKASI PEKERJAAN BERDASARKAN NILAI MATA KULIAH (STUDI KASUS: ALUMNI PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI FAKULTAS ILMU KOMPUTER UNIVERSITAS BRAWIJAYA)".