 ISSN:2302-8149
e-ISSN:2540-9719

Sentiment Analysis of User Perceptions Toward the Gemini Application using the Naïve Bayes Algorithm

[1]Cynthia Hayat*
[1]Program Studi Sistem Informasi, Fakultas Teknologi Cerdas, Center for Knowledge Management and Collaborative Innovation, Universitas Kristen Krida Wacana
[1]Jl.Tanjung Duren Raya No.4, Jakarta Barat, 11470, Indonesia
*e-mail: cynthia.hayat@ukrida.ac.id

(***received*:** 12 November 2025, ***revised*:** 15 April 2026, ***accepted*:** 11 July 2026)

Abstract

Sentiment analysis, a pivotal technique in Natural Language Processing (NLP), facilitates the classification of textual data into positive or negative sentiment categories. This study applies a Naïve Bayes classification model to analyse 1,000 user reviews of the Gemini application, an AI-powered assistant developed by Google AI, collected from the Google Play Store. The research adopts the Knowledge Discovery in Database (KDD) methodology, encompassing data selection, preprocessing, transformation, data mining, and evaluation stages. Reviews were pre-processed using text normalization techniques and transformed into numerical vectors using Term Frequency–Inverse Document Frequency (TF-IDF). The sentiment classifier achieved a high accuracy of 94.57%, with a precision of 95.7% and an F1-score of 96.7%, indicating effective classification performance. The results revealed a predominance of positive sentiment, with frequently occurring terms such as "assist," "good," and "application." Conversely, negative sentiments were often associated with issues regarding language support and system limitations. These findings provide actionable insights into user satisfaction and areas for improvement in the Gemini application. While the Naïve Bayes algorithm proved efficient and interpretable, limitations include difficulty handling nuanced linguistic features such as sarcasm or ambiguity. Future research may enhance sentiment classification performance by incorporating larger datasets and advanced models such as BERT or deep neural networks. Overall, this study demonstrates the practical utility of machine learning-based sentiment analysis in guiding user-centric application development.

Keywords: gemini application, naïve bayes, sentiment analysis, text classification

1 Introduction

Sentiment analysis is a key technique within the field of Natural Language Processing (NLP) that aims to determine whether a given piece of text expresses a positive or negative sentiment [1]. This method is widely used to understand user preferences and perceptions toward various products or services. The typical sentiment analysis workflow includes stages such as text preprocessing, data transformation, and model evaluation [2] [3].

The advancement of artificial intelligence (AI) and NLP technologies in recent years has significantly reshaped user interactions with digital platforms. AI-powered tools are now capable of understanding natural language inputs, generating relevant responses, and adapting to user feedback. A prominent example is Gemini, an AI assistant developed by Google AI and officially launched in 2024. Gemini utilizes NLP to support a variety of user tasks, such as answering questions, generating code, and analysing images.

To refine performance, Gemini integrates a built-in feedback mechanism that enables users to rate responses as either "Good" or "Bad" and add optional comments. While this internal feedback provides valuable insights, it captures only a fraction of user sentiment. Publicly available reviews, such as those on the Google Play Store, represent a richer and more diverse source of user opinion. These reviews can be systematically analysed to uncover patterns in user satisfaction, usability, and perceived shortcomings of the application[4], [5].

Due to the unstructured and large-scale nature of such reviews, sentiment analysis emerges as an effective approach to transforming qualitative user feedback into actionable insights. Machine

 ISSN:2302-8149
e-ISSN:2540-9719

learning techniques, particularly text classification algorithms like Naïve Bayes, are commonly employed in this context due to their efficiency and accuracy. Prior research has demonstrated the effectiveness of Naïve Bayes in sentiment classification tasks, particularly in domains involving large datasets and noisy textual input [6], [7].

This study implements a Naïve Bayes-based sentiment classification model to analyse user reviews of the Gemini application collected from the Google Play Store. The objective is to classify these reviews into positive or negative categories and assess the overall public perception of the app. The findings are expected to provide evidence-based insights that can inform future improvements to Gemini and contribute to user-centric design enhancements [8], [9].

The importance of this research is underscored by the growing accessibility of information in the digital era, driven by innovations in search engines and AI technologies. Google, a leader in the technology sector, has been instrumental in these advancements through platforms such as Google Search and its AI-focused division, Google AI. As a result, applications like Gemini have been developed to facilitate real-time, intelligent interactions with users.

Gemini leverages NLP to deliver contextually appropriate responses and supports additional features like code generation and image analysis. The app's continuous improvement strategy involves a feedback loop through which users can rate system responses and offer suggestions. This user-driven evaluation is essential for enhancing application quality and maintaining high levels of user engagement and trust [10]. In addition to internal feedback mechanisms, over 1.3 million user reviews on the Google Play Store—accompanied by an average rating of 4.7—represent a valuable data source for deeper sentiment analysis (Google Gemini – Apps on Google Play, n.d.). These reviews reflect a broad range of user experiences, which are analysed in this study using the Naïve Bayes classifier, a method that has proven effective in previous sentiment analysis applications [10].

Several relevant studies support this methodological choice. For instance, [11] applied sentiment analysis to reviews of Google Classroom, achieving 82% accuracy and identifying a predominance of negative sentiment, signalling areas for improvement. Similarly, [12] examined reviews of the Kinemaster application, reporting an 85% accuracy rate with mostly positive but also considerable negative feedback.

Moreover, [13] investigated user sentiment toward the BRImo application and achieved 84.52% accuracy, again observing a positive sentiment majority alongside notable criticisms. [14] analysed sentiment toward the Gojek app using Naïve Bayes and recorded 68% accuracy, with critical feedback pointing to user difficulties and technical issues.

As user experience becomes a key differentiator among competing apps, understanding user sentiment is critical for improving platform features, fostering trust, and driving long-term engagement. User-generated reviews on app stores and online platforms represent a valuable yet underutilized source of insights into user perceptions, expectations, and pain points. These reviews often contain rich emotional and experiential information that can inform data-driven improvements. Despite this potential, there remains a notable lack of comprehensive machine learning-based analyses of user sentiment specifically focused on the Gemini App.

Building on these precedents, the present study applies the Naïve Bayes algorithm to classify user sentiment regarding Gemini. The resulting insights will support strategic improvements to the app, ensuring that future updates align more closely with user needs and expectations, and contribute to a more seamless and satisfying user experience. As user experience becomes a key differentiator among competing apps, understanding user sentiment is critical for improving platform features, fostering trust, and driving long-term engagement. User-generated reviews on app stores and online platforms represent a valuable yet underutilized source of insights into user perceptions, expectations, and pain points. These reviews often contain rich emotional and experiential information that can inform data-driven improvements. Despite this potential, there remains a notable lack of comprehensive machine learning-based analyses of user sentiment specifically focused on the Gemini App.

The Naïve Bayes algorithm offers a simple, interpretable, and computationally efficient approach for text classification tasks such as sentiment analysis. However, its application to the analysis of user reviews for the Gemini App has received little attention in the academic and professional literature. Addressing this gap could provide both methodological insights and practical recommendations for stakeholders involved.

 ISSN:2302-8149
e-ISSN:2540-9719

This study differs from previous research as it specifically analyzes user perceptions of the Gemini application using a Naïve Bayes approach based on publicly available review data from the Google Play Store. The scientific contributions of this study are twofold: theoretically, it validates the performance of the Naïve Bayes algorithm in a new domain of AI-powered assistant applications, demonstrating its robustness in handling real-world user-generated text data, and practically, it provides actionable insights for the Gemini development team to improve user experience, particularly in enhancing language-related features and addressing user-reported limitations.

The objectives of this research are to develop and implement a Naïve Bayes-based sentiment analysis model to classify user reviews of the Gemini app into positive and negative categories using publicly available review data, and to identify key sentiment trends and actionable insights regarding user satisfaction, common issues, and feature preferences, with the goal of informing future improvements to the Gemini app's user experience.

2 Literature Review

2.1 Sentiment Analysis and Naïve Bayes

Sentiment analysis is a technique in Natural Language Processing (NLP) that aims to identify whether a sentiment contains positive, negative, or neutral elements (Aftab et al., 2023). Sentiment analysis can also be utilized to determine user preferences toward a particular product or service. The sentiment analysis process typically consists of several stages, including text preprocessing, data transformation, and evaluation [12], [14]

Several methods can be employed to perform sentiment analysis; however, this study applies the Naïve Bayes algorithm. The Naïve Bayes approach is a probabilistic model that generates predictions using available evidence and prior probability distributions. It is widely utilized in text classification tasks due to its simplicity, computational efficiency, and strong performance with large textual datasets. In this research, the algorithm is used to classify user reviews as either positive or negative sentiments.

The Naïve Bayes algorithm is a supervised classification method grounded in Bayes' Theorem, which predicts the probability of an observation belonging to a specific class based on a given set of features [8], [15]. Despite its conceptual simplicity, Naïve Bayes remains one of the most effective and widely used algorithms for text classification and other probabilistic learning tasks.

At its core, the algorithm relies on the principle of conditional probability, where the likelihood of a class given certain features is calculated from prior knowledge and observed data. It operates under the class-conditional independence assumption, meaning that each feature contributes independently to the probability of a particular class, regardless of the presence or absence of other features [6], [16]. This assumption enables the model to achieve high computational efficiency and robustness, even when dealing with large and high-dimensional datasets.

2.2 Related Works

Several studies have explored the use of machine learning techniques for sentiment analysis, with Naïve Bayes (NB) emerging as one of the most popular algorithms due to its simplicity and effectiveness in text classification tasks. Early works by [17] demonstrated that Naïve Bayes, when combined with feature engineering techniques such as term frequency–inverse document frequency (TF-IDF), could achieve competitive performance in movie review sentiment classification. Subsequent research by [18] further refined NB approaches by incorporating techniques like feature selection and smoothing to handle data sparsity and improve classification accuracy.

Prior research highlights the growing potential of user-generated reviews as valuable sources for sentiment analysis, offering insights into user satisfaction and guiding product development. Various studies have utilized natural language processing (NLP) techniques to pre-process and classify review data, transforming unstructured text into meaningful sentiment categories. Naïve Bayes classifiers, particularly the Multinomial variant, have proven effective for text classification tasks due to their simplicity and robustness in handling word frequency-based features. Complementary techniques such as case folding, tokenization, stop word removal, and stemming are commonly applied to normalize text data before classification. Web Mining, including automated scraping, has also become a widely adopted method to collect large-scale review data from online platforms, enabling scalable

 ISSN:2302-8149
e-ISSN:2540-9719

sentiment analysis workflows. Recent advances in natural language processing, including word embedding's and deep learning techniques, have also been integrated with NB frameworks to enrich feature representations [19]. These developments underscore the continued relevance and adaptability of Naïve Bayes-based approaches in the evolving landscape of sentiment analysis research.

Building on this foundation, the present study as research position that focuses on applying the Multinomial Naïve Bayes algorithm for binary sentiment classification of user reviews specifically for the Gemini application. Manual labelling of review data as positive or negative enables supervised learning, ensuring model reliability. The study employs Google Colab for model training and performance evaluation using a confusion matrix, achieving an impressive accuracy of 94.57% and strong precision, recall, and F1-scores. Furthermore, word frequency analysis supplements classification results by revealing commonly used positive expressions such as "assist" and "good", providing additional interpretability of user sentiment. These results confirm prior findings regarding the effectiveness of Naïve Bayes for sentiment analysis and demonstrate its applicability for understanding user feedback in the context of mobile applications.

3 Research Method

The data used in this research will be secondary data, specifically sourced from user reviews and ratings of the Gemini application on the Google Play Store. A total of 1,000 individual reviews and their corresponding application ratings will be selected for analysis. These reviews will include both textual feedback and numerical ratings provided by users of the Gemini app.

To collect the data, the study will utilize Web Mining techniques, which allow for the automated extraction of data from online sources. Specifically, the google-play-scraper library, a Python-based tool designed for scraping data from the Google Play Store, will be employed. This library facilitates the retrieval of various app-related data, including review content, star ratings, review timestamps, and reviewer information.

The data collection process will involve the following steps:

- Scraping Process: Using the google-play-scraper library, the program will query the Google Play Store for reviews related to the Gemini app.
- Data Extraction: Relevant data points, such as the text of user reviews, ratings (ranging from 1 to 5 stars), and the date of submission, will be extracted.
- Data Preprocessing: The collected data will undergo preprocessing steps such as cleaning (removal of special characters, links, and irrelevant information) and normalization (standardizing text format) to ensure consistency and facilitate analysis.

This methodology ensures that the dataset is comprehensive, structured, and suitable for sentiment analysis using machine learning techniques, providing an in-depth understanding of user opinions and feedback on the Gemini application.

3.1 Research Procedurs

The methodology used in this study is Knowledge Discovery in Database (KDD), which consists of five main stages [20], [21]. The first stage, Selection, involves choosing relevant data for analysis. In the Preprocessing stage, the selected data is processed by cleaning and normalizing it to ensure its quality. Then, in the Transformation stage, the data is converted into a more useful format for further analysis. Data Mining is the core stage, where analytical techniques are applied to extract patterns or insights from the data. Finally, the Interpretation/Evaluation stage focuses on interpreting the results of the analysis and evaluating the relevance and accuracy of the findings.

This KDD process ensures that the data used in the study is properly processed, leading to the discovery of valuable information that can be applied in decision-making. Each stage is interconnected to optimize the outcomes of the data analysis.

 ISSN:2302-8149
e-ISSN:2540-9719

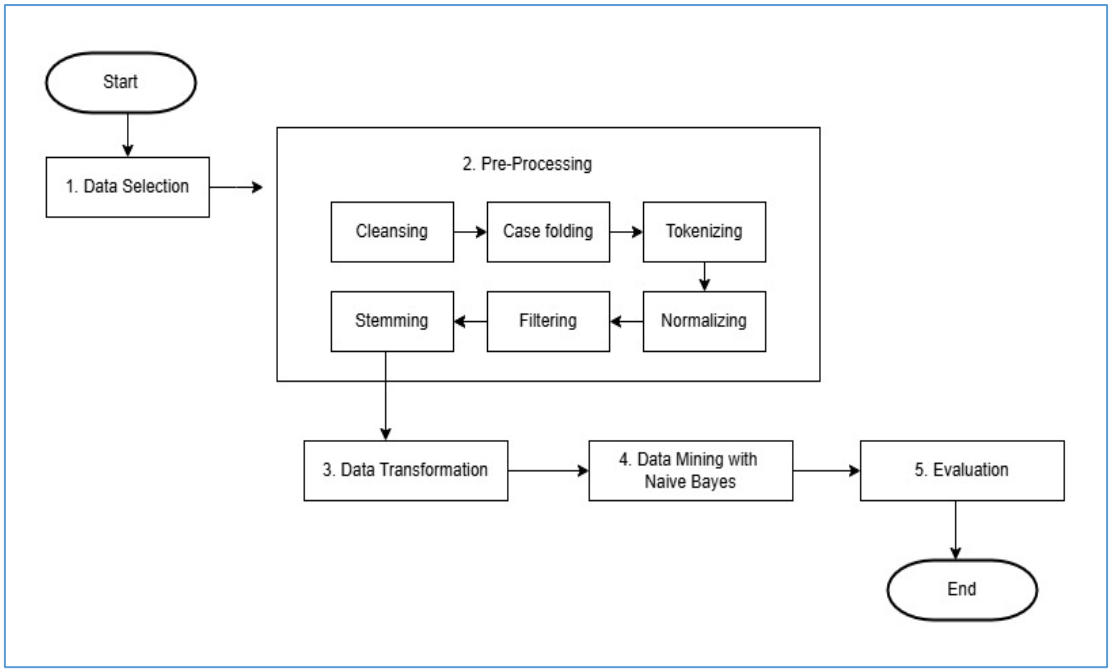

Figure 1 Procedures of analyizing user sentiment with TF-IDF

This study comprises several key stages in Figure 1, each integral to the overall sentiment analysis process:

Phase 1: Data Selection

Data selection refers to the automated extraction of large volumes of data from various online sources. This process is essential for gathering high-value textual content, which can be analysed to uncover patterns, trends, or insights relevant to decision-making in a business context. Next step is the labelling phase, the collected review data is categorized based on sentiment polarity. Ratings of 4 and 5 are assigned a positive label, while ratings of 1 and 2 are considered negative. Reviews with a rating of 4 and 5 are assigned a positive label, while ratings of 1 and 2 are considered negative. Reviews with a rating less than or equal to 2 are labeled as negative, while those with a rating greater than or equal to 4 are labeled as positive. Reviews with a rating of 3 are not assigned any label and are given a null value, as they represent neutral or ambiguous sentiments. Reviews with a rating of 3 were categorized as neutral and excluded from the analysis. A total of 60 neutral reviews (6% of the dataset) were removed to ensure clear class separation and reduce labeling ambiguity. Neutral sentiment often contains mixed or unclear polarity, which can negatively affect classification performance, particularly in probabilistic models such as Naïve Bayes. This exclusion strategy is consistent with prior sentiment analysis studies, where neutral data are removed to enhance model discriminative capability and reduce noise in binary classification tasks.

Phase 2: Text Preprocessing

After removing neutral and unlabeled data, the final dataset consisted only of positive and negative classes, ensuring a clear binary classification setup. Text preprocessing is a crucial step to enhance the quality and consistency of the input data. This includes the removal of unlabelled entries, followed by a series of text normalization techniques: case folding (converting text to lowercase), stop word removal, tokenization, and stemming. These steps help reduce noise and improve the reliability of sentiment detection. The dataset is divided into training and testing subsets using an 80:20 ratio. The training data (80%) is used to build the model, while the test data (20%) is utilized to evaluate its performance.

Phase 3: Data Transformation

This stage involves the application of Term Frequency–Inverse Document Frequency (TF-IDF) weighting to transform textual data into numerical feature vectors. TF-IDF helps quantify the importance of words within a document relative to the entire dataset, thereby enhancing the model's

 ISSN:2302-8149
e-ISSN:2540-9719

ability to detect sentiment-laden terms. Python libraries such as TfidfVectorizer and CountVectorizer are employed in this process.

Phase 4: Data Mining
The sentiment classification model was developed using the Naïve Bayes algorithm, implemented through the scikit-learn (version 1.x) library in Python 3.10, running on the Google Colab environment. This configuration ensures computational efficiency, reproducibility, and compatibility with modern NLP workflows.
The pseudocode of the implemented Multinomial Naïve Bayes algorithm is presented below:

Algorithm 1: Sentiment Classification using Naïve Bayes
Input: Preprocessed text data (X), Sentiment labels (Y)
Output: Predicted sentiment classes
- Split dataset into training (X_train, Y_train) and testing (X_test, Y_test)
- Transform text into numerical features using TF-IDF vectorization
- Initialize Naïve Bayes classifier: NB = MultinomialNB()
- Train model: NB.fit(X_train, Y_train)
- Predict on test set: Y_pred = NB.predict(X_test)
- Evaluate model using confusion matrix and metrics (Accuracy, Precision, Recall, F1)
- Return performance metrics and classification results

This pseudocode represents the core classification process used in this research and can be replicated by other researchers under the same computational environment.

Phase 5: Model Evaluation and Visualization
The final phase involves evaluating the classification model using a confusion matrix, which reports performance in terms of True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). This analysis enables a clear understanding of the model's accuracy in identifying sentiment orientation. Additionally, text visualizations are generated to highlight the most frequent and influential terms in both positive and negative reviews, providing further insights into sentiment trends.

To ensure model robustness and generalizability, this study applies k-fold cross-validation (k=5). The dataset is partitioned into five subsets, where each fold is used once as testing data while the remaining folds are used for training. This approach provides a more reliable estimate of model performance compared to a single train-test split, reducing the risk of overfitting and performance bias.

4 Results and Analysis

4.1 Data Preparation and Text PreProcessing

This section describes the preparatory stages of data collection and preprocessing before conducting the sentiment classification experiment. The process includes three main steps: data scraping, labeling, and text preprocessing. Each stage was designed to ensure that the dataset was accurate, representative, and suitable for text mining and machine learning analysis.

- Data Scrapping

The first step in this research involved data acquisition through web scraping. The google-play-scraper Python library was used to automatically collect 1,000 user reviews of the Gemini application from the Google Play Store. This library provides a structured interface to extract app-related information, such as review text, star ratings, review date, and reviewer metadata. The scraping process was conducted within the Google Colab environment to facilitate reproducibility and ensure stable network access. Data were retrieved in JSON format and converted into a structured dataset using the Pandas library. The collected data were saved in CSV format for further preprocessing. The result of this process can be seen in Figure 2 which displays a portion of the extracted user reviews, including columns for review text, rating score, and timestamp.

 ISSN:2302-8149
e-ISSN:2540-9719

	reviewId	userName	userImage	content	score	thumbsUpCount	reviewCreatedVersion	at	replyContent	repliedAt	appVersion
0	9e83c38b-98b0-40f0-9140-b65216ae895f	Iswandi ST	https://play-lh.googleusercontent.com/a/ACg8oc...	Sejauh ini bagus, tetapi ada yang kurang: -tol...	5	378	1.0.668480831	2024-10-20 12:03:15	None	None	1.0.668480831
1	0a172ec8-56cb-44e5-92db-962eb57d2c98	Riky	https://play-lh.googleusercontent.com/a-/ALV-U...	Masih banyak kurangnya. Tidak bisa di perintah...	2	0	1.0.668480831	2024-11-03 13:27:36	None	None	1.0.668480831
2	3bf98a16-ac7a-41d2-a861-fa541abb6f2f	Keysa Putri larisa	https://play-lh.googleusercontent.com/a-/ALV-U...	So far apk ini bagus bgt!! . Tapi ada beberapa...	5	1399	1.0.667532867	2024-09-13 06:52:17	None	None	1.0.667532867
3	1e149397-f02a-4d46-9537-6ef6f1abe9b3	YESSY ADILLA	https://play-lh.googleusercontent.com/a/ACg8oc...	Aplikasinya sudah bagus dan mudah untuk diguna...	5	310	1.0.668480831	2024-10-17 13:09:47	None	None	1.0.668480831
4	f7aecd5d-8375-4edd-a342-378c1ba4fc0a	BN	https://play-lh.googleusercontent.com/a-/ALV-U...	Model bahasa AI setelah di update semakin buru...	1	57	1.0.668480831	2024-10-13 13:03:23	None	None	1.0.668480831

Figure 2 Result of data scraping

- Labeling

Labeling is the next stage after completing the data scraping process. In this stage, the rating or score data are used as a reference to assign positive and negative sentiment labels. Reviews with a rating less than or equal to 2 are labeled as negative, while those with a rating greater than or equal to 4 are labeled as positive. Reviews with a rating of 3 are not assigned any label and are given a null value, as they represent neutral or ambiguous sentiments.

content	score	Label
Kenapa tidak bisa terbuka saat dipakai	4	Positif
sangat membantu dan simpel	5	Positif
Cukup membantu. AI bisa dibilang cukup pintar, tetapi masih kalah dibandingkan dengan pesaingnya. Reasoning terkadang masih kurang, masih sering halu, belum bisa dibuat asisten untuk melakukan penelitian.	3	
Bagus, multifungsi, keren, dan canggih!	5	Positif
aplikasi ini sangat membantu saya untuk mencari informasi, dengan waktu saya yg begitu singkat dalam sehari-hari yg selalu di habiskan dengan kesibukan sehari-hari	4	Positif
aplikasi ai pertama yang aku pakai dan aku suka❤️ semua bisa di lakuin sama ai ini, mulai dari merubah tulisan menjadi teks, merangkum video YouTube, pokonya banyak deh, hemat memori juga, simple, dan yang pasti free	5	Positif
Aplikasi harian paling penting si, kadang kan kita nyari informasi di google atau yutub ga ada karena masalahnya sangat spesifik tapi kalo nanya disini pasti ada jawabannya, the best pokoknya wajib download sih	5	Positif
Tiap buka aplikasi harus di apdèt ulang, maknya saya kasih 3	3	
Ai bodoh ini saya menanyakan pertanyaan seperti cerita,hal²,dll. Tapi dia gampang lupa ingatan dan juga konteksnya ke mana².jadi harus memberi pertanyaannya yang super detail karena AI ini gampang lupa ingatan dan konteksnya ke mana²	1	Negatif
tidak jelas sangat tidak membantu	1	Negatif

Figure 3 Result of labeling

- Text Processing

After the labeling process, the data were examined to identify whether any null values were present. During this inspection, 60 records containing null values were detected and After completing the inspection, all records with null values underwent a data cleaning process and were subsequently removed from the dataset. Once the null entries were deleted, the dataset was rechecked to ensure that no missing values remained, as shown in Table 1.

Table 1 Null data before and after data cleaning

Before Data Cleaning	Content	Score	Label
	0	0	60
After Data Cleaning	Content	Score	Label
	0	0	0

- Case folding

After the cleaning process, the data proceeded through a sequence of text preprocessing stages: case folding, stopword removal, tokenizing, and stemming. During the case folding step, all characters in the dataset were converted to lowercase, as illustrated in Figure 4.

	content	score	Label	text_clean
0	sangat berguna untuk terjemahan dan sangat mud...	5	Positif	sangat berguna untuk terjemahan dan sangat mud...
1	berguna, fleksibel,mudah dipahami, gratis, rinci	5	Positif	berguna fleksibelmudah dipahami gratis rinci
2	ini aplikasi AI gratis yang menurutku sudah bagus	5	Positif	ini aplikasi ai gratis yang menurutku sudah bagus
3	aplikasi yang sangat membantu di dalam segala ...	5	Positif	aplikasi yang sangat membantu di dalam segala ...
4	Sangat membantu dalam belajar	5	Positif	sangat membantu dalam belajar
5	Jika memang Gemini dikembangkan, maka hapus ap...	1	Negatif	jika memang gemini dikembangkan maka hapus apl...
6	aplikasinya bagus sangat berguna untuk memdapa...	5	Positif	aplikasinya bagus sangat berguna untuk memdapa...
7	enak di ajak bicara tentang informasi tertentu...	5	Positif	enak di ajak bicara tentang informasi tertentu...
8	Yo aplikasi ini sangat membantu untuk belajar ...	5	Positif	yo aplikasi ini sangat membantu untuk belajar ...
9	bagus aplikasinya bisa membantu dengan mudah	4	Positif	bagus aplikasinya bisa membantu dengan mudah

Figure 4 Result of case folding

 ISSN:2302-8149
e-ISSN:2540-9719

- Stopword Removal

After all characters in the dataset were converted to lowercase, the next cleaning stage was stopword removal, which eliminates common words that appear frequently but carry little or no meaningful information. This process also removes symbols, punctuation marks, and other unnecessary entities, as shown in Figure 5.

	content	score	Label	text_clean	text_StopWord
0	sangat berguna untuk terjemahan dan sangat mud...	5	Positif	sangat berguna untuk terjemahan dan sangat mud...	berguna terjemahan mudah
1	berguna, fleksibel,mudah dipahami, gratis, rinci	5	Positif	berguna fleksibelmudah dipahami gratis rinci	berguna fleksibelmudah dipahami gratis rinci
2	ini aplikasi AI gratis yang menurutku sudah bagus	5	Positif	ini aplikasi ai gratis yang menurutku sudah bagus	aplikasi ai gratis menurutku bagus
3	aplikasi yang sangat membantu di dalam segala ...	5	Positif	aplikasi yang sangat membantu di dalam segala ...	aplikasi membantu mantap
4	Sangat membantu dalam belajar	5	Positif	sangat membantu dalam belajar	membantu belajar
5	Jika memang Gemini dikembangkan, maka hapus ap...	1	Negatif	jika memang gemini dikembangkan maka hapus apl...	gemini dikembangkan hapus aplikasi asisten goo...

Figure 5 Result of stopword removal

- Tokenizing

The subsequent cleaning step after removing stopwords is tokenizing, which involves splitting the text into smaller units or tokens so that it can be analyzed more effectively. This process is illustrated in Figure 6.

	content	score	Label	text_clean	text_StopWord	text_tokens
0	sangat berguna untuk terjemahan dan sangat mud...	5	Positif	sangat berguna untuk terjemahan dan sangat mud...	berguna terjemahan mudah	[berguna, terjemahan, mudah]
1	berguna, fleksibel,mudah dipahami, gratis, rinci	5	Positif	berguna fleksibelmudah dipahami gratis rinci	berguna fleksibelmudah dipahami gratis rinci	[berguna, fleksibelmudah, dipahami, gratis, ri...
2	ini aplikasi AI gratis yang menurutku sudah bagus	5	Positif	ini aplikasi ai gratis yang menurutku sudah bagus	aplikasi ai gratis menurutku bagus	[aplikasi, ai, gratis, menurutku, bagus]
3	aplikasi yang sangat membantu di dalam segala ...	5	Positif	aplikasi yang sangat membantu di dalam segala ...	aplikasi membantu mantap	[aplikasi, membantu, mantap]
4	Sangat membantu dalam belajar	5	Positif	sangat membantu dalam belajar	membantu belajar	[membantu, belajar]

Figure 6 Result of tokenizing

- Stemming

Once the text has been divided into tokens, the final cleaning stage, stemming, is performed. Stemming reduces each word to its base or root form, ensuring that different variations of the same word are treated uniformly. The result of this process is shown in Figure 7.

	content	score	Label	text_clean	text_StopWord	text_tokens	text_steamindo
0	sangat berguna untuk terjemahan dan sangat mud...	5	Positif	sangat berguna untuk terjemahan dan sangat mud...	berguna terjemahan mudah	[berguna, terjemahan, mudah]	guna terjemah mudah
1	berguna, fleksibel,mudah dipahami, gratis, rinci	5	Positif	berguna fleksibelmudah dipahami gratis rinci	berguna fleksibelmudah dipahami gratis rinci	[berguna, fleksibelmudah, dipahami, gratis, ri...	guna fleksibelmudah paham gratis rinci
2	ini aplikasi AI gratis yang menurutku sudah bagus	5	Positif	ini aplikasi ai gratis yang menurutku sudah bagus	aplikasi ai gratis menurutku bagus	[aplikasi, ai, gratis, menurutku, bagus]	aplikasi ai gratis turut bagus
3	aplikasi yang sangat membantu di dalam segala ...	5	Positif	aplikasi yang sangat membantu di dalam segala ...	aplikasi membantu mantap	[aplikasi, membantu, mantap]	aplikasi bantu mantap
4	Sangat membantu dalam belajar	5	Positif	sangat membantu dalam belajar	membantu belajar	[membantu, belajar]	bantu ajar
5	Jika memang Gemini dikembangkan, maka hapus ap...	1	Negatif	jika memang gemini dikembangkan maka hapus apl...	gemini dikembangkan hapus aplikasi asisten goo...	[gemini, dikembangkan, hapus, aplikasi, asiste...	gemini kembang hapus aplikasi asisten google b...

Figure 7 Result of Stemming

 ISSN:2302-8149
e-ISSN:2540-9719

4.2 Data Transformation

After all data had gone through the text preprocessing stage, the next step was data splitting. The dataset was divided into training data and testing data with a ratio of 80:20, where 80% of the data were used for training and 20% were used for testing.

Following the data splitting process, the next step was data transformation using the Term Frequency–Inverse Document Frequency (TF-IDF) weighting method. This transformation was applied to convert textual data into numerical feature representations that could be processed by the Naïve Bayes algorithm during model training. The TF-IDF weighting was implemented automatically using Python's Scikit-learn library, particularly the TfidfVectorizer and CountVectorizer functions.

After the data were transformed, they were used to train the Naïve Bayes classification model. The training process was performed automatically using the Scikit-learn library, specifically the MultinomialNB module. Once the model was trained using the training data, it was then evaluated using the testing data to measure its performance and validate the accuracy of the sentiment classification results.

4.3 Classification Performance

Upon completion of the testing phase, the data were evaluated using a confusion matrix, which produced several key performance metrics, including accuracy, precision, recall, and F1-score.

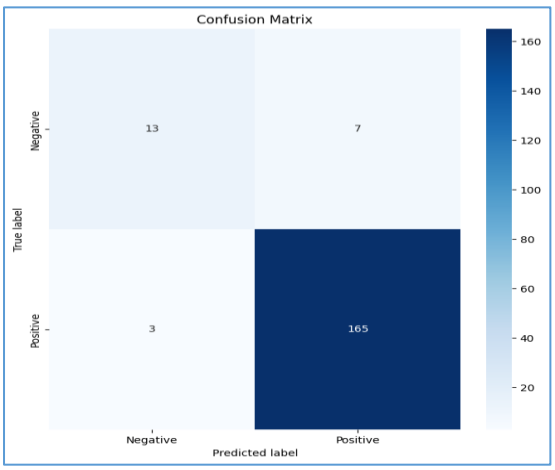

Figure 8 Confusion matrix

The analysis of the confusion matrix for the Naïve Bayes model in Figure 2 illustrates 165 True Positives, 13 True Negatives, 7 False Positives, and 3 False Negatives. Based on these values, the model demonstrated an accuracy of 94.57%, a precision of 95.7%, a recall of 98.21%, and an F1-score of 96.7%. These results indicate that the Gemini application is predominantly perceived positively by users, as reflected in the high number of positive reviews. The high precision and recall values suggest that the model is effectively identifying positive sentiment. However, the presence of a small proportion of negative sentiment, as indicated by the False Positives and False Negatives, highlights areas for potential improvement in the application's functionality and user experience. This suggests that while the application is performing well, further refinement is warranted to optimize its performance and address the concerns raised in the negative feedback.

	precision	recall	f1-score	support
Negatif	0.81	0.65	0.72	20
Positif	0.96	0.98	0.97	168
accuracy			0.95	188
macro avg	0.89	0.82	0.85	188
weighted avg	0.94	0.95	0.94	188

Figure 9 Model performance metrics

 ISSN:2302-8149
e-ISSN:2540-9719

These results in Figure 9 indicate that the Naïve Bayes algorithm is highly effective for sentiment classification tasks, particularly in differentiating positive and negative user opinions. The high precision and recall scores show that the model can accurately detect positive sentiments while minimizing misclassification.

To further interpret the sentiment distribution, the analysis revealed that positive sentiments dominated the dataset, with keywords such as *"assist," "good,"* and *"application"* appearing most frequently. In contrast, negative sentiments were often associated with terms like *"language," "Gemini,"* and *"Google."*

4.4 Class Imbalance Analysis

The dataset exhibits a significant class imbalance, with approximately 85% positive and 15% negative reviews. This imbalance may bias the model toward the majority class, potentially inflating accuracy while reducing sensitivity to minority (negative) sentiment. Despite achieving high accuracy (94.57%), this result should be interpreted with caution, as accuracy alone may not adequately reflect model performance in imbalanced datasets. In such cases, evaluation metrics such as precision, recall, and F1-score provide a more reliable assessment of model effectiveness, particularly in identifying minority class instances.

The cross-validation results demonstrate consistent model performance, with an average accuracy of approximately 94% ± 1.0, indicating that the Naïve Bayes classifier maintains stability across different data splits. This low variation suggests that the model generalizes well across different data partitions.

Table 2 5-Fold cross-validation results

Fold	Accuracy	Precision	Recall	F1-Score
Fold-1	94.12	95.10	97.85	96.45
Fold-2	93.68	94.85	97.40	96.10
Fold-3	94.89	96.02	98.10	97.05
Fold-4	94.21	95.33	97.65	96.47
Fold-5	93.70	94.90	97.30	96.08
Average	**94.12**	**95.24**	**97.66**	**96.43**
Std Dev	**0.44**	**0.45**	**0.33**	**0.39**

The 5-fold cross-validation results demonstrate consistent model performance across different data splits. The Naïve Bayes classifier achieved an average accuracy of 94.12% ± 0.44, with precision, recall, and F1-score values of 95.24%, 97.66%, and 96.43%, respectively. The low standard deviation across all evaluation metrics indicates that the model exhibits minimal performance variation, suggesting strong robustness and generalizability in classifying user sentiment.

4.5 Sentimen Distribution

Sentiment distribution refers to the categorization and representation of sentiments expressed in a given dataset, typically in the form of positive, negative, and neutral sentiments. It is a crucial step in sentiment analysis, as it shows how the opinions or emotions of individuals are distributed across different sentiments in the data being analyzed. In a typical sentiment analysis process, sentiment distribution could be represented visually using charts or graphs (such as pie charts or bar graphs), where:

- Positive Sentiment: Indicates a favourable or approving opinion.
- Negative Sentiment: Reflects dissatisfaction or disapproval.

 ISSN:2302-8149
e-ISSN:2540-9719

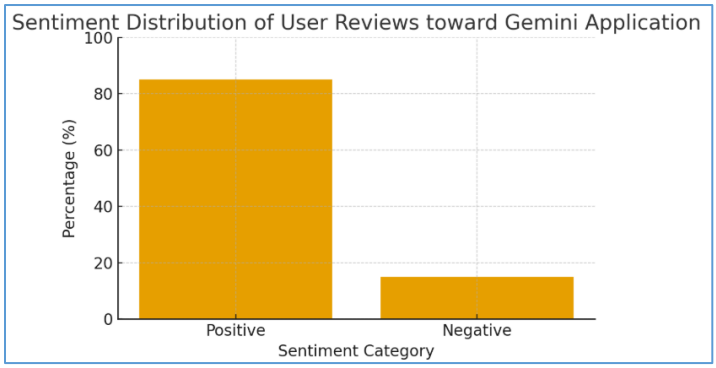

Figure 10 Sentiment distribution of user reviews toward gemini application

Figure 10 illustrates the proportion of positive and negative sentiments expressed by users in their reviews of the Gemini application. The majority of users expressed positive sentiments (approximately 85%), indicating a generally favorable perception of the application's functionality and usefulness. Meanwhile, negative sentiments (around 15%) highlight areas that require improvement, particularly related to language features and system limitations.

To identify the positive and negative words found in user sentiment, a visualization of these words will be presented in the form of a table as shown in table 3.

Table 3 The positive and negative sentiment

Words	Positive Sentiment		Words	Negative Sentiment	
	Freq	%		Freq	%
assist	331	5.36	language	60	3.72
good	253	4.09	gemini	41	2.54
application	245	3.97	google	38	2.36
gemini	169	2.74	AI	34	2.11
his/her/its	118	1.91	application	30	1.86
easy	104	1.68	his/her/its	26	1.61
lesson	100	1.62	indonesia	22	1.36
AI	98	1.59	not	21	1.30
search	93	1.51	nor	20	1.24
google	86	1.39	assist	16	0.99

Based on Table 2, positive sentiment is predominantly associated with terms such as *"bantu"* (assist), *"bagus"* (good), and *"aplikasi"* (application), indicating that users perceive the Gemini application as beneficial and of high quality. Conversely, negative sentiment is frequently characterized by words such as *"bahasa"* (language), *"Gemini"*, and *"Google"*, suggesting that users express criticism toward the language features of the Gemini application, a product developed by Google.

These findings indicate that the Naïve Bayes algorithm is effective in sentiment classification, demonstrating a high level of accuracy in distinguishing between positive and negative user reviews. Nevertheless, this study is not without limitations. The current model could be enhanced by expanding the dataset size and employing alternative machine learning algorithms. Such improvements are expected to further increase the robustness and generalizability of the sentiment analysis framework presented in this research.

4.6 Interpretation of Result

The findings of this study offer valuable insights into the development of the Gemini application, particularly in understanding user perceptions and preferences. The dominance of positive sentiment in the dataset suggests a potential bias in user-generated reviews, which may influence the

 ISSN:2302-8149
e-ISSN:2540-9719

classification outcomes. This phenomenon is common in real-world datasets, where users are more likely to submit positive feedback. As a result, the model may become less sensitive to negative sentiment, which is critical for identifying areas of improvement in the application**.**

The predominance of words such as *"helpful"*, *"good"*, and *"application"* in the positive sentiment cluster indicates that users perceive Gemini as a useful and functional tool. This suggests that developers should maintain and further enhance the features that users find beneficial, especially in terms of usability and accessibility. On the other hand, the frequent appearance of terms like *"language"*, *"Gemini"*, and *"Google"* in the negative sentiment cluster highlights user concerns regarding the language features or linguistic performance of the application. These findings imply that developers should evaluate and improve language-related functionalities, such as translation accuracy, multilingual options, and personalization features, to boost overall user satisfaction.

The classification model employed in this study, Naïve Bayes with Term Frequency–Inverse Document Frequency (TF-IDF) weighting demonstrated reliable performance in sentiment classification based on textual reviews. One of the primary strengths of this model is its computational efficiency and its ability to handle large datasets quickly, making it a viable solution for real-world industrial applications. However, the model's limitations include a reduced capacity to interpret complex semantic structures such as sarcasm, irony, or word ambiguity. Additionally, the use of only two sentiment labels (positive and negative) constrains the range of user expression, excluding neutral feedback. Thus, although the model is effective in terms of basic accuracy and efficiency, there remains significant room for improvement through the integration of context-aware models like BERT or neural network-based architectures.

Methodologically, this study supports previous research findings that affirm the effectiveness of probabilistic algorithms in text classification tasks. For instance, from previous studies demonstrated that Naïve Bayes, when paired with proper preprocessing and feature engineering, can yield competitive performance. However, this study takes a more simplified approach compared to others that utilize multi-category classification or deep learning models based on transformers. In this context, the present study offers a lightweight yet effective alternative and provides a solid empirical foundation for future exploration into hybrid models that combine computational efficiency with deeper semantic understanding.

The results demonstrate that the Naïve Bayes algorithm, when combined with TF-IDF feature extraction, performs reliably in classifying user reviews of the Gemini application. The model's accuracy of 94.57% surpasses several previous studies on sentiment analysis using Naïve Bayes. For instance, [12] reported 82% accuracy on Google Classroom reviews, [13] achieved 85% for Kinemaster, and [14] obtained 84.52% for BRImo app reviews. Similarly, [15] reached 68% accuracy for Gojek sentiment analysis.

Compared to these studies, the present research shows a higher performance level, suggesting that the Gemini dataset is more semantically consistent or that the preprocessing and feature engineering steps (e.g., TF-IDF) were more effective in capturing sentiment-bearing words. The simplicity and efficiency of Naïve Bayes also contribute to the model's strong generalization across textual variations.

However, the model still exhibits minor misclassifications, particularly in detecting complex expressions such as sarcasm or context-dependent phrases. This limitation reflects a common challenge in probabilistic models that rely solely on word frequency without deeper semantic understanding.

Overall, the findings confirm that the Naïve Bayes algorithm provides a lightweight yet accurate solution for sentiment classification of user-generated reviews. It also highlights valuable insights for Gemini developers particularly the need to enhance language support and reduce linguistic-related issues reported by users.

5 Conclusion

Based on the research, the use of the Naïve Bayes algorithm has proven to yield a high accuracy of 94.57%, indicating that sentiment analysis is well-suited for this algorithm. Additionally, the confusion matrix results show 165 True Positives, 13 True Negatives, 7 False Positives, and 3 False Negatives. These findings suggest that the Gemini application is predominantly perceived positively,

 ISSN:2302-8149
e-ISSN:2540-9719

which is further supported by the visualization of dominant words such as "help" (appearing 331 times) and "good" (appearing 253 times). Although Gemini is largely perceived positively, there are still some negative sentiments toward the application, highlighting potential areas for improvement and further development to enhance the user experience. Future research is recommended to explore deep learning-based models such as BERT to enhance the ability to detect sarcasm and to expand the analysis to include reviews written in the Indonesian language.

Future work may address class imbalance by applying advanced resampling techniques such as Synthetic Minority Over-sampling Technique (SMOTE), undersampling, or class weighting to improve the model's ability to detect minority class instances more effectively.

Reference

[1] W. Medhat, A. Hassan, and H. Korashy, "*Sentiment Analysis Algorithms and Applications: A Survey*," *Ain Shams Eng. J.*, Vol. 5, No. 4, 2014, DOI: 10.1016/j.asej.2014.04.011.

[2] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "*A Survey on Sentiment Analysis Methods, Applications, and Challenges*," *Artif. Intell. Rev.*, Vol. 55, No. 7, 2022, DOI: 10.1007/s10462-022-10144-1.

[3] D. M. E. D. M. Hussein, "*A Survey on Sentiment Analysis Challenges*," *J. King Saud Univ. - Eng. Sci.*, Vol. 30, No. 4, 2018, DOI: 10.1016/j.jksues.2016.04.002.

[4] C. Villavicencio, J. J. Macrohon, X. A. Inbaraj, J. H. Jeng, and J. G. Hsieh, "*Twitter Sentiment Analysis Towards Covid-19 Vaccines in the Philippines using Naïve Bayes*," *Inf.*, Vol. 12, No. 5, 2021, DOI: 10.3390/info12050204.

[5] A. Ligthart, C. Catal, and B. Tekinerdogan, "*Systematic Reviews in Sentiment Analysis: A Tertiary Study*," *Artif. Intell. Rev.*, Vol. 54, No. 7, 2021, DOI: 10.1007/s10462-021-09973-3.

[6] C. Burlăcioiu, C. Boboc, B. Mirea, and I. Dragne, "*Text Mining in Business. A Study of Romanian Client's Perception with Respect to using Telecommunication and Energy Apps*," *Econ. Comput. Econ. Cybern. Stud. Res.*, Vol. 57, No. 1, 2023, DOI: 10.24818/18423264/57.1.23.14.

[7] H. Uğuz, "*Detection of Carotid Artery Disease by using Learning Vector Quantization Neural Network*," *J. Med. Syst.*, Vol. 36, No. 2, 2012, DOI: 10.1007/s10916-010-9498-8.

[8] Z. Li, R. Li, and G. Jin, "*Sentiment Analysis of Danmaku Videos based on Naïve Bayes and Sentiment Dictionary*," *IEEE Access*, Vol. 8, 2020, DOI: 10.1109/ACCESS.2020.2986582.

[9] M. B. Ressan and R. F. Hassan, "*Naïve-Bayes Family for Sentiment Analysis during COVID-19 Pandemic and Classification Tweets*," *Indones. J. Electr. Eng. Comput. SCI.*, Vol. 28, No. 1, 2022, DOI: 10.11591/ijeecs.v28.i1.pp375-383.

[10] N. Rane, S. Choudhary, and J. Rane, "*Gemini Versus ChatGPT: Applications, Performance, Architecture, Capabilities, and Implementation*," *SSRN Electron. J.*, 2024, DOI: 10.2139/ssrn.4723687.

[11] D. Nurwahidah, G. Dwilestari, N. Dienwati Nuris, and R. Narasati, "Analisis Sentimen Data Ulasan Pengguna Aplikasi *Google* Kelas pada *Google Play Store* menggunakan *Algoritma Naïve Bayes*," *JATI (Jurnal Mhs. Tek. Inform.*, Vol. 7, No. 6, 2024, DOI: 10.36040/jati.v7i6.8245.

[12] K. Kevin, M. Enjeli, and A. Wijaya, "Analisis Sentimen Pengunaaan Aplikasi Kinemaster menggunakan Metode *Naive Bayes*," *J. Ilm. Comput. SCI.*, Vol. 2, No. 2, 2024, DOI: 10.58602/jics.v2i2.24.

[13] M. K. K. Insan, U. Hayati, and O. Nurdiawan, "Analisis Sentimen Aplikasi Brimo pada Ulasan Pengguna di *Google Play* menggunakan *Algoritma Naive Bayes*," *JATI (Jurnal Mhs. Tek. Inform.*, Vol. 7, No. 1, 2023, DOI: 10.36040/jati.v7i1.6373.

[14] K. D. Indarwati and H. Februariyanti, "Analisis Sentimen Terhadap Kualitas Pelayanan Aplikasi Gojek menggunakan Metode *Naive Bayes Classifier*," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, Vol. 10, No. 1, 2023, DOI: 10.35957/jatisi.v10i1.2643.

[15] A. P. Pimpalkar and R. J. Retna Raj, "*Influence of Pre-Processing Strategies on the Performance of ML Classifiers Exploiting TF-IDF and BOW Features*," *ADCAIJ Adv. Distrib. Comput. Artif. Intell. J.*, Vol. 9, No. 2, 2020, DOI: 10.14201/adcaij2020924968.

[16] I. F. Rozi, E. N. Hamdana, M. Balya, and I. Alfahmi, "Pengembangan Aplikasi Analisis

Sentimen *Twitter* (Studi Kasus Samsat Kota Malang)," *Inform. Polinema*, Vol. 1, No. 1, 2018.

[17] S. Jiang, G. Pang, M. Wu, and L. Kuang, "*An Improved K-Nearest-Neighbor Algorithm for Text Categorization*," *Expert Syst. Appl.*, Vol. 39, No. 1, 2012, DOI: 10.1016/j.eswa.2011.08.040.

[18] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell, "*Text Classification from Labeled and Unlabeled Documents using EM*," *Mach. Learn.*, Vol. 39, No. 2, 2000, DOI: 10.1023/a:1007692713085.

[19] F. Ahmad, X. W. Tang, J. N. Qiu, P. Wróblewski, M. Ahmad, and I. Jamil, "*Prediction of Slope Stability using Tree Augmented Naive-Bayes classifier: Modeling and Performance Evaluation*," *Math. Biosci. Eng.*, Vol. 19, No. 5, 2022, DOI: 10.3934/mbe.2022209.

[20] S. Głowania, J. Kozak, and P. Juszczuk, "*Knowledge Discovery in Databases for a Football Match Result*," *Electron.*, Vol. 12, No. 12, 2023, DOI: 10.3390/electronics12122712.

[21] A. M. Abdulkadium, R. A. A. Shekan, and H. A. Hussain, "*Application of Data Mining and Knowledge Discovery in Medical Databases*," *Webology*, Vol. 19, No. 1, 2022, DOI: 10.14704/web/v19i1/web19329.