Published online on the journal's webpage: **http://journal.ittelkom-pwt.ac.id/index.php/dinda**

Journal of Dinda

Data Science, Information Technology, and Data Analytics

Vol. 4 No. 2 (2024) 115 - 122 E-ISSN: 2809-8064

Development of Sentiment Analysis System of Simple Pol Application on Google Play Store Using Naive Bayes Classifier Method and BERT Prediction

Muhammad Dhito Maulidan[1*], Sri Sumarlinda[2], Sopingi[3]

[1*,2,3]Department of Informatics Engineering, Duta Bangsa University Surakarta

[1*]dhito.maulidan@gmail.com, [2]srisumarlinda@udb.ac.id, [3]sopingi@udb.ac.id

Abstract

Digitalization in public services raises various sentiments that are very dynamic, one example is the Simpel Pol Health Test application made by PT Cipta Sari Arsonia (CSA). The research objective is to obtain useful information from accurate community review sentiments for service improvement and feedback for service providers and application developers. The method used is Naïve Bayes Classifier with Tf-idf weighting, Multinomial Naïve Bayes with review value indicators and review sentences predicted by the BERT method as a determinant of sentiment value whether positive or negative. Sentiment towards the application shows quite encouraging results, from 3000 data analyzed with 1772 positive reviews and 263 negative reviews, 80% training data and 20% test data, the naïve bayes classification model is able to provide a high level of accuracy by giving a balanced performance of the two methods, which is 88.7% with a precision of 88.5%, recall of 100% and f1-score of 93.9%. The data showed that most people gave a positive response to this application, with the dominant word being 'easy'. This system was developed using the local-based streamlit framework and proved to be quite reliable in developing systems for data processing and web-based data analysis even though the scraping process is slightly longer than the google colab service. Future research is expected to be able to predict data that is positive or negative with several parameters and several sentiment analysis methods at once and their comparison.

Keywords: *BERT, Driver License Health Test, Google Playstore, Naïve Bayes, Streamlit*

1. Introduction

In recent years, the process of renewing driving licenses (SIM) in Indonesia has undergone several changes aimed at improving service quality and accessibility. However, some issues still need attention, such as the high issuance fees and the complexity of the renewal process. One potential solution to these problems is the Simpel Pol application developed by PT Cipta Sari Arsonia (CSA).

This application facilitates the public in applying for and renewing driving licenses online, in accordance with the applicable regulations. According to Article 81, Paragraph 4 of Law No. 22 of 2009, to obtain a driving license (SIM), every individual is required to meet health requirements, including physical health certified by a doctor and mental health proven by passing a psychological test [1]. This provision is reinforced by the Indonesian Police Regulation No. 9 of 2012, Article 36, which states that the mental health aspects tested for obtaining a driving license include the ability to concentrate, accuracy, self-control, adaptability, emotional stability, and work endurance [2].

The use of digital technology in the SIM renewal service not only helps reduce costs and time but also facilitates the public in the application process. This research aims to provide a deeper understanding of the benefits and challenges of implementing online SIM renewal services. This service not only saves applicants' queue time but also encourages the public to adapt to the digitalization era. By continuously optimizing the use of digital technology across various sectors, including public services like SIM renewal, Indonesia can accelerate its digital transformation and improve the overall quality of life for its citizens.

To delve deeper into the online SIM renewal service provided by the government through the Simpel Pol application, this research will investigate the reliability and speed of public services, which are expected to

Received: 26-07-2024 | Accepted: 04-08-2024 | Published: 08-08-2024

improve significantly with sentiment analysis. Sentiment analysis, also known as opinion mining, is a computational study aimed at recognizing and expressing opinions, sentiments, evaluations, attitudes, emotions, subjectivity, judgments, or views contained in a text [3].

In this research, the sentiment analysis method using the Naïve Bayes algorithm will be employed to evaluate the effectiveness of services provided by the government, particularly in West Java and Central Java. According to several studies, Naïve Bayes has several advantages, including computational speed, algorithmic simplicity, and high accuracy. Accurate sentiment analysis results will serve as an evaluation for relevant stakeholders, enabling them to improve the quality of services provided.

Additionally, this research also applies prediction using the BERT (Bidirectional Encoder Representations from Transformers) method, which is relatively new and innovative. This study further develops a system based on the Python framework, Streamlit, an open-source Python-based framework designed to simplify the development of interactive web applications in the field of data science and machine learning [4].

This research is expected to provide valuable insights for relevant stakeholders in enhancing the quality of online SIM renewal services. Overall, this study serves as a first step in exploring the potential of online SIM issuance and renewal services through the Simpel Pol application. By utilizing sentiment analysis methods and developing a system based on the Python framework, it is hoped that the research findings will contribute positively to efforts to improve public service feedback in Indonesia. The ongoing digital transformation in Indonesia will become increasingly beneficial if public services can be optimized effectively through the appropriate use of technology.

2. Research Methods

In this chapter, the research methodology is divided into several stages to facilitate understanding of the research and will be designed in the following flowchart:

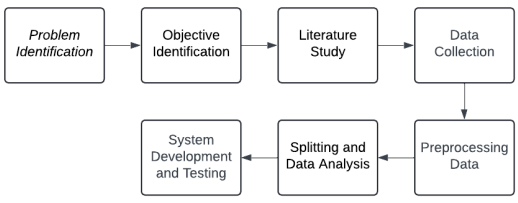


Figure 1. Research Flow Scheme

2.1 *Problem Identification*

The author conducted observations and identified issues within the Simpel Pol application service by analyzing user comments and reviews on the Google Play Store. Through this analytical method, the author was able to pinpoint user-facing problems and assess the quality of the provided service. This process involves detailed monitoring of user feedback to gain a comprehensive understanding of existing shortcomings. By employing this approach, the author aims to provide a clear depiction of areas that need improvement in the Simpel Pol application.

2.2 *Objective Identification*

At this stage, the author explains the framework and objectives of the research. In Chapter 1, it was stated that the objective of this research is to analyze and classify user reviews of the Simpel Pol application on Google Playstore using the Naïve Bayes method, which will then be developed into a practical system.

2.3 *Literature Study*

The aim of this stage is to identify relevant theories that will be used to address the research problems, as well as to obtain strong references as a basis for the researcher. These references can come from various sources such as books, other research studies, and articles on the internet.

2.4 *Data Collection*

The data used in this research consists of primary data from reviews of the Simpel Pol application on Google Playstore and several secondary data sources from research journals, articles, or books from the internet that are relevant to the research theme. The method used to collect data from application reviews is web scraping, conducted with Python and facilitated by Google Colab to expedite the data retrieval process. The Python programming language library or package used is google_play_scraper, which, by simply inputting the application package name from Google Playstore, can efficiently extract various string-formatted data that can be converted into spreadsheet tables, typically in CSV format.

2.5. *Preprocessing Data*

Data preprocessing is a crucial step conducted prior to the classification process, aimed at transforming the data into a form that is more suitable for the Naive Bayes algorithm. In research, various issues such as missing values, redundant data, or inappropriate data formats often hinder the outcomes of the data mining process. To address these challenges, preprocessing is necessary. This stage involves a series of steps to clean the data, making it ready for analysis. The goal of this step is to optimize the data used in the classification process, ensuring that the results obtained are more accurate and

reliable, thereby enhancing the overall reliability of the data analysis [5]. Below are some of the preprocessing steps:

a. Text cleaning is the first step after inputting the dataset. The next step is to broadly clean the data by deleting unused columns, removing data that only contains emojis, or eliminating unnecessary punctuation marks such as exclamation marks, at (@) symbols, and other similar characters from each comment and review.
b. Case folding is a step in text processing that aims to convert each word into a uniform format, specifically lowercase. This process is carried out using the lower method on strings in Python. The primary purpose of case folding is to ensure that all words in the text data are processed in a consistent form, namely lowercase, to minimize unnecessary variations in text analysis [6].
c. Stopword removal is the process of filtering out insignificant words from the results of tokenization, with the aim of selecting important words that represent the document [7]. Stopwords are eliminated from the text to reduce noise and allow the analysis to focus more on relevant words.
d. Tokenizing: In the tokenization process, each word is split based on the presence of spaces between them, as an initial step that defines the boundaries between smaller linguistic units in the analyzed text.
e. Stemming: In this research, the stemming process is used as a method to convert words with affixes into their base forms. This technique utilizes the Sastrawi library to perform this transformation, enabling a more in-depth analysis of the structure and meaning of the analyzed text.
f. Labeling, Comments are labeled as follows: scores of 1 to 2 are categorized as negative, scores of 4 to 5 as positive, and a score of 3 as neutral. These scores are directly obtained from scraping reviews of the Simpel Pol application from the Google Play Store using the google_play_scraper library, typically found in a column named ‘score’. Subsequently, neutral-labeled data will be translated and predicted using the BERT (Bidirectional Encoder Representations from Transformers) method via the Pipeline sentiment-analysis, initialized with the transformer’s library. This library is commonly used in research for Language Modeling, Question Answering, Machine Translation, and Text Summarization.

2.6 *Splitting and Data Analysis*

The labeled data is then divided into two parts: training data and test data. Naive Bayes utilizes Bayes' Theorem, a straightforward mathematical formula used to calculate conditional probability. Conditional probability measures the likelihood of an event occurring given that another event has already occurred. This can include assumptions, hypotheses, statements, or evidence. One advantage of this algorithm is that it only requires training data to determine the parameters needed for the classification process [8]. The general formula is expressed in Equation (1).

$$P(A|B) = \frac{P(B|A)\cdot P(A)}{P(B)} \quad (1)$$

Description:

P (A|B) = conditional probability of A given B.

P (B|A) = conditional probability of B given A.

P (A) = probability of event A.

P (B) = probability of event B.

While calculating accuracy, precision, recall, and f1-score equations (2), (3), (4), (5).

$$\text{accuracy} = \frac{\text{Number of right predictions}}{Total\ amount\ of\ predictions} \quad (2)$$

$$\text{precision} = \frac{\text{True Positive (TP)}}{\text{True Positive(TP)+False Positive (FP)}} \quad (3)$$

$$\text{recall} = \frac{\text{True Positive (TP}}{\text{True Positive (TP)+False Negative (FN)}} \quad (4)$$

$$\text{F1-Score} = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision+Recall}} \quad (5)$$

2.7 *Splitting and Data Analysis*

In this final step, the researcher embarked on developing a sentiment analysis system utilizing the Naive Bayes method. This development was carried out using the Streamlit framework, which resulted in the creation of a web application specifically designed for sentiment analysis. The application was built using the Python programming language, allowing the implementation of models from the fields of machine learning and data science [9].

In addition to developing the system, the researcher conducted extensive testing using several sample datasets to evaluate the system's effectiveness and reliability. The process, including the methodology and the steps taken, will be comprehensively illustrated

in the flowchart presented below.

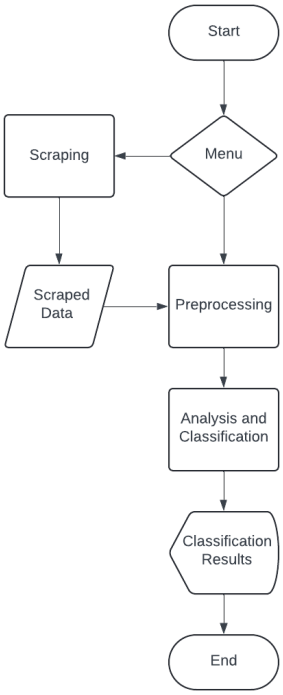


Figure 2. Schematic of System Development Flow

The output of the created system includes a table containing several columns with preprocessing results, evaluation results such as the number of data points (positive and negative), a confusion matrix, evaluation metrics (classification evaluation), and a word cloud.

3. Results and Discussion

From the above research methodology, this study can be summarized with the following key results and discussion points: data or dataset scraping, data preprocessing, data division and analysis, all developed with a web-based system framework (Streamlit).

3.1 *Dataset Scraping*

Data collection was conducted through a process known as web scraping, which is a technique for automatically extracting information from a website. In this case, data was gathered from the Google Play Store using the Python programming language, with the assistance of the `google_play_scraper` library. This API library facilitates the extraction of application information and reviews from the Google Play Store without relying on third-party or external dependencies [10]. The data was collected from the Simpel Pol application package with the ID `com.ngi.sim`. The platform used for data extraction was Google Colab, which allows for efficient cloud-based data storage.

The dataset scraping process takes a few minutes, but it can be faster with a stable internet connection. By using Google Colab as the data collection platform, data storage is expected to be more efficient because it uses cloud storage provided by the platform. Below are the results of the dataset scraping using Google Colab:

47	7109a02e-d36c-4c5a-865d-9baaf0e0605b	Nevynandika Ibram	https://play-lh.googleusercontent.com/a-/ALV-U...	aplikaai pelayanan publik verifikasi aja nungg...	1	20
...	...	...	...	...	...	...
2024	da22b38b-09f1-43ca-9eeb-745762c69c5b	kyo nichi	https://play-lh.googleusercontent.com/a-/ALV-U...	aplikasi nya bagus dan simple	5	0
2496	da22b38b-09f1-43ca-9eeb-745762c69c5b	kyo nichi	https://play-lh.googleusercontent.com/a-/ALV-U...	aplikasi nya bagus dan simple	5	0
2020	c6476119-fd34-4cfd-a77e-383b9be6ddab	Zavidca Nirvansyam	https://play-lh.googleusercontent.com/a/ACg8oc...	mudah banget.....selanjutnya apa.....? m...	5	0
2495	c6476119-fd34-4cfd-a77e-383b9be6ddab	Zavidca Nirvansyam	https://play-lh.googleusercontent.com/a/ACg8oc...	mudah banget.....selanjutnya apa.....? m...	5	0
2494	42a41201-68a1-4497-8d18-ef2dbbb5eca1	Suparti Mulyono	https://play-lh.googleusercontent.com/a-/ALV-U...	Wah bagus	5	0

2994 rows × 13 columns

Figure 3. Simple Pol Scarping with Google Colab

The successfully exported data will be processed using the Streamlit framework to facilitate usage and create a simpler interface. This research aims to create data scraping code from Google Colab that can be easily used by laypeople. By using Streamlit, the interface is expected to be more user-friendly and efficient. The ultimate goal of developing this system is to create data scraping code that can be used by a wide range of users, including laypeople, with a more intuitive and simpler interface.

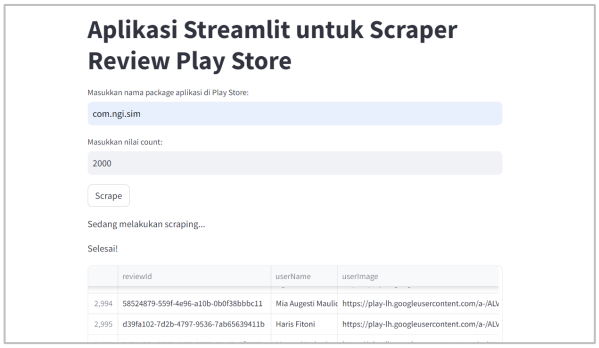


Figure 4. Streamlit Scraping data play store

The dataset consists of 2999 entries, starting from index 0, resulting in a total of 3000 rows of data. The column names include username, reviewerid, content or comment, review rating, and review date. With the content or comment column, the research can proceed to data preprocessing.

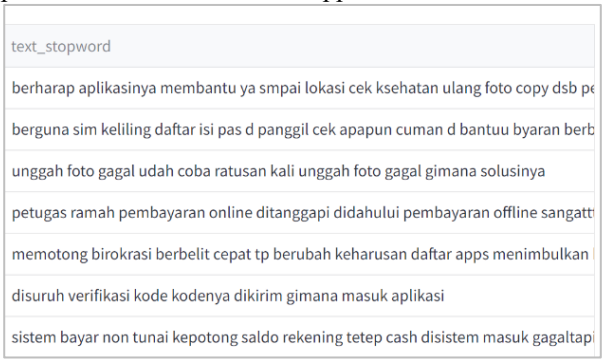
3.2 *Dataset Cleaning and Preprocessing*

The data preprocessing stage in this research begins with cleaning the data from duplicates, null values, or empty entries. Additionally, the data will be converted to lowercase for consistency in the analysis. Emoticons will be removed, and duplicates will be eliminated to make the data more concise and freer from missing values. The cleaned comment data will be stored in the text_clean column as follows:

	komentar	text_clean
2,978	Bagus, sangat membantu	bagus sangat membantu
2,979	Bagus banget praktis	bagus banget praktis
2,981	Baru mencoba tapi lumayan agak bingung	baru mencoba tapi lumayan agak bingung
2,983	Alhamdulillah,bisa mempermudah kami	alhamdulillahbisa mempermudah kami
2,984	Bagus banget sangat membantu	bagus banget sangat membantu
2,985	Aplokasinya mendukung banget bagi yg lom tau caranya.....	aplokasinya mendukung banget bagi yg lom tau caranya
2,986	Simpel bener dan sangat membantu	simpel bener dan sangat membantu
2,987	Mempercepat proses pembuatan dan perpanjang SIM	mempercepat proses pembuatan dan perpanjang sim
2,988	oke dan praktis	oke dan praktis

Figure 5. Streamlit text cleaning

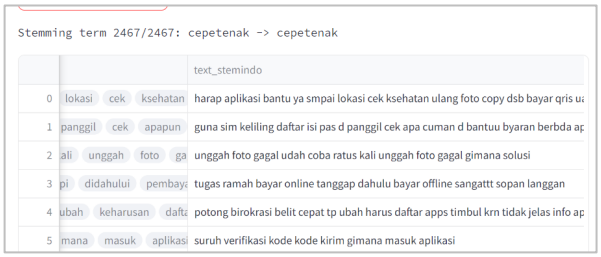
Data that has undergone the cleaning process will be processed using the Stopword Removal method to eliminate unnecessary words in the processing. The use of the NLTK library with a preference for Indonesian is expected to enhance the effectiveness of filtering out unnecessary words. The Natural Language Toolkit (NLTK) is a Python-based platform developed for processing text data, including tasks such as stemming, classification, tokenization, parsing, and tagging [11]. Additionally, the researcher has incorporated several abbreviations and non-standard words, such as "yg," "ga," and "dgn," to ensure a more effective word filtering process. The results will appear as shown below.



text_stopword
berharap aplikasinya membantu ya smpai lokasi cek ksehatan ulang foto copy dsb pe
berguna sim keliling daftar isi pas d panggil cek apapun cuman d bantuu byaran berb
unggah foto gagal udah coba ratusan kali unggah foto gagal gimana solusinya
petugas ramah pembayaran online ditanggapi didahului pembayaran offline sangatt
memotong birokrasi berbelit cepat tp berubah keharusan daftar apps menimbulkan
disuruh verifikasi kode kodenya dikirim gimana masuk aplikasi
sistem bayar non tunai kepotong saldo rekening tetep cash disistem masuk gagaltapi

Figure 6. Streamlit stopwords removal

After stopword removal, the text will be divided into tokenization. Tokenization is the process of separating each word in a sentence and at the same time removing certain characters that are considered punctuation marks (Asiyah, 2016). After tokenization, the next step required is stemming with Indonesian language preference. For this, a Python library called Sastrawi is used. The process in this library can generate the base word in a sentence by removing prefixes, suffixes, inserts, or a combination of the three, also known as affixes.



Stemming term 2467/2467: cepetenak -> cepetenak

		text_stemindo
0	lokasi cek ksehatan	harap aplikasi bantu ya smpai lokasi cek ksehatan ulang foto copy dsb bayar qris u
1	panggil cek apapun	guna sim keliling daftar isi pas d panggil cek apa cuman d bantuu byaran berbda a
2	ali unggah foto ga	unggah foto gagal udah coba ratus kali unggah foto gagal gimana solusi
3	pi didahului pembay	tugas ramah bayar online tanggap dahulu bayar offline sangattt sopan langgan
4	ubah keharusan daft	potong birokrasi belit cepat tp ubah harus daftar apps timbul krn tidak jelas info a
5	mana masuk aplikasi	suruh verifikasi kode kode kirim gimana masuk aplikasi

Figure 7. Streamlit tokenization and stemming

3.3. *Sentiment Prediction*

After preprocessing, including data cleaning and stemming, the data will undergo prediction and be labeled in the label column according to the prediction rules outlined in the following table.

Table 1. Table software and supporting hardware

Review Score	Label Review	Description
1-2	Negative	Data has negative sentiment
3	Neutral	Data has neutral sentiment and will be predicted
4-5	Positive	Data has positive sentiment

In research, the accuracy of the data utilized is crucial to ensuring the validity of the obtained results. One method to minimize data uncertainty is through prediction. Sentiment analysis labeled as neutral is conducted using BERT (Bidirectional Encoder Representations from Transformers), a transformer model developed by Google, which aims to understand the bidirectional context of text. Unlike previous transformer models that process text in a single direction (either left to right or right to left), BERT processes text in both directions (left to right and right to left) simultaneously. This method is implemented in the sentiment analysis pipeline of the Transformers library, focusing on the use of the BERT model for sentiment prediction. A key advantage of this method is its ability to determine sentences with ambiguous sentiment. For instance, negative reviews may arise from misunderstandings, while positive words may carry irony. In this context, the BERT-based approach represents a recent innovation. The language model developed by Google's AI, BERT, possesses the capability to analyze words, understand word relationships, and provide a deeper contextual understanding of sentences [12].

Before making predictions with BERT to enhance accuracy, researchers also translate the text into English

for better recognition by machine learning models. The translation is performed using the deep_translator package from GoogleTranslator, a free Python library that offers high flexibility and unlimited usage. This library facilitates text translation between languages in a straightforward manner and supports all languages.

Features include the ability to translate words from files, obtain translation results from various sources, and automatically detect languages. With its high-level abstraction and support for multiple languages, the library is user-friendly. Additionally, its API is designed for ease of use, and the library is regularly maintained to ensure stability in its application [13]. The prediction process outlined above yields results as follows:

nilai	text_english	label
3	I'll try going to a doctor's office. Is it really practical?	negatif
3	Cannot enter place of birth other than Indonesian territory	negati
3	Where is the doctor's address in Surakarta city? It doesn't men	negatif
3	Good	positif
3	help	positif
3	Can't install, the app is not friendly with old devices	negatif

Figure 8. Streamlit prediction.

The number of data points labeled as positive and negative is displayed in the system as follows:

Jumlah Data Positif dan Negatif

	Sentiment	Jumlah
0	Positif	1772
1	Negatif	263

Figure 9. Streamlit number of positively and negatively labeled data

After the final preprocessing stage, which involves labeling, the dataset contains 1,772 entries labeled as positive and 263 entries labeled as negative. The data is now ready for analysis using the Naive Bayes classification.

Table 2. Data Splitting

	Data Train	Data Test
Positives	1418	354
Negatives	210	53

The first step is to determine TF-IDF by extracting features using the TF-IDF (Term Frequency-Inverse Document Frequency) method. The primary goal of the TF-IDF method is to evaluate the importance of a term within a document relative to a broader collection of documents [14]. Additionally, this method transforms text into numerical vectors that can be understood by machine learning models.

Once the features are extracted, the Naive Bayes model is trained using the training data that has been converted into TF-IDF vectors. The extracted data is then used by the Naive Bayes model for multiclass classification, employing the multinomial Naive Bayes method. Multinomial Naive Bayes is a specialized variant of the Naive Bayes method utilized in text mining for text classification, leveraging class probabilities within documents based on word frequency [15]. This method yields a MultinomialNB score of 0.8869778869778869 (0.89).

The next step is to determine the metrics of the confusion matrix. The Confusion Matrix is a technique used to compute the values of precision, recall, and accuracy. Typically, the values obtained from the confusion matrix are expressed as percentages (%) [16]. By utilizing the confusion matrix, we can assess the performance of the model in predicting the given data.

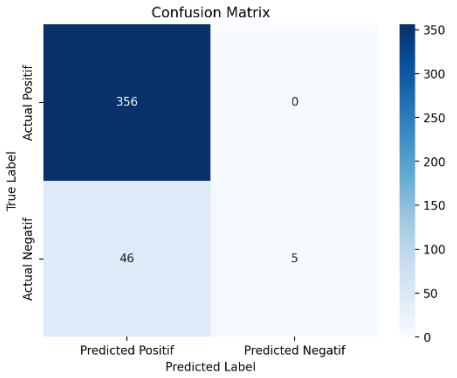


Figure 10. Streamlit Confusion Matrix result.

Next, evaluation metrics are determined, which include a classification report for Naive Bayes that provides measures such as accuracy, precision, recall, and F1-score

Accuracy: 0.8869778869778869

Precision: 0.8855721393034826

Recall: 1.0

F1-Score: 0.9393139841688655

Classification Report:

	precision	recall	f1-score	support
negatif	1.00	0.10	0.18	51
positif	0.89	1.00	0.94	356
accuracy			0.89	407
macro avg	0.94	0.55	0.56	407
weighted avg	0.90	0.89	0.84	407

Figure 11 Streamlit evaluation metrics.

From the training data, there are 1,418 positive entries and 210 negative entries, while the test data includes 3,554 positive entries and 53 negative entries. The model achieved an accuracy of 88.7%, precision of 88.5%, recall of 100%, and an F1-score of 93.9%. The model's results are also visualized with a word cloud, which provides an overview of the main themes or issues in the text, with more frequently occurring words represented larger and more prominently.

Figure 12 Streamlit wordcloud visualization.

3.5 *System Development*

The final step is to develop a web-based system using the Streamlit framework. This research system is designed with a one-click analysis feature, allowing for the import of CSV/XLSX files to generate tables that can also be exported. The one-click analysis feature encompasses several processes, including preprocessing, analysis, evaluation of results, and visualization. With multiple trials and calibrations, the system has been refined to be robust and capable of executing various sentiment analysis steps. Users can simply drag and drop the datasheet and click to analyze

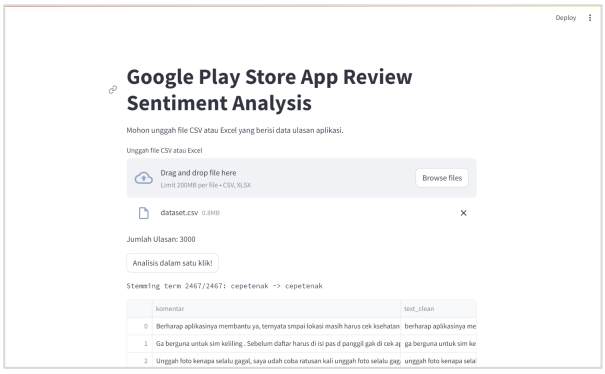


Figure 13 Streamlit sentiment analysis of play store app review.

4. Conclusion

Based on the research results from 3,000 data points from the health test application for driving license registration or renewal, the Naïve Bayes classification model proved to be quite accurate, with 1,772 positive labels (87.1%) and 263 negative labels (12.9%). The training data proportion was 0.8 (80%), and the testing data proportion was 0.2 (20%), resulting in an accuracy rate of 88.7% and a precision rate of 88.5%.

In terms of prediction, the researcher concluded that the combination of the lightweight and fast Naïve Bayes model with the high-accuracy but somewhat resource-intensive BERT prediction provides a balanced performance. This combination effectively leverages the strengths of both methods.

Regarding system development, a few minor issues were encountered, such as the longer scraping test times using Streamlit in a local environment compared to Google Colab or the cloud. Additionally, the time required for stemming in Streamlit was comparable to that of Google Colab.The researcher hopes that future studies will prioritize various sentiment analysis methods with comparative results, aiming to achieve stronger accuracy and precision and deliver the best performance.

References

[1] S. N. RI, Undang-Undang Rpublik Indonesia Nomor 22 Tahun 2009 Tentang Lalu Lintas dan Angkutan Jalan, Jakarta, 2009.

[2] K. K. N. R. Indonesia, Peraturan Kepala Kepolisian Negara Republik Indonesia Nomor 9 Tahun 2012 Tentang Surat Izin Mengemudi, Jakarta, 2012.

[3] Shahariar, G. M., Biswas, S., Omar, F., Shah, F. M., & Binte Hassan, S. (2019). Spam Review Detection Using Deep Learning. 2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference, IEMCON 2019, 27–33.

[4] Amal, M. "Membuat Aplikasi Web Sains Data dengan Mudah Menggunakan Streamlit" 15 March 2021. [Online]. Available at: https://informatics.uii.ac.id/2021/03/15/streamlit-membuat-aplikasi-web-sains-data. [Acessed 7 August 2024].

[5] Muttaqin, F. A., & Bachtiar, A. M. (2016). Implementasi Teks Mining Pada Aplikasi Pengawasan Penggunaan Internet Anak "Dodo Kids Browser." Jurnal Ilmiah Komputer Dan Informatika, 1–8.

[6] M. U. Albab, Y. Karuniawati P, and M. N. Fawaiq, "Optimization of the Stemming Technique on Text preprocessing President 3 Periods Topic," J. Transform., vol. 20, no. 2, pp. 1–10, 2023, [Online]. Available: https://journals.usm.ac.id/index.php/transformatika/page1.

[7] M. S. Anwar, I. M. I. Subroto, and S. Mulyono, "Sistem Pencarian E-Journal Menggunakan Metode Stopword Removal Dan Stemming Berbasis Android," Konf. Ilm. Mhs. Unissula 2, pp. 58–70, 2019.

[8] R. F. D. Pratiwi, S. Sumarlinda, and F. E. Nastiti, "Comparative Analysis of Restock Needs Bottled Water Using K-Nearest Neighbor (K-NN), Support Vector Machine (SVM), and the Naïve Bayes Algorithm," Int. J. Inf. Syst. Technol. Data Sci., vol. 1, no. 1, pp. 1–8, 2023.

[9] M. Ferdyandi, N. Y. Setiawan, and F. Abdurrachman Bachtiar, "Prediksi Potensi Penjualan Makanan Beku Berdasarkan Ulasan Pengguna Shopee Menggunakan Metode Decision Tree Algoritma C4.5 Dan Random Forest (Studi Kasus Dapur Lilis)," J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 6, no. 2, pp. 588–596, 2022, [Online]. Available: http://j-ptiik.ub.ac.id.

[10] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest," … Teknol. Inf. dan …, vol. 6, no. 9, pp. 4305–4313, 2022, [Online]. Available: http://j-ptiik.ub.ac.id.

[11] A. E. Budiman and A. Widjaja, "Analisis Pengaruh Teks Preprocessing Terhadap Deteksi Plagiarisme Pada Dokumen Tugas Akhir," J. Tek. Inform. dan Sist. Inf., vol. 6, no. 3, pp. 475–488, 2020.

[12] A. A. Mudding, "Mengungkap Opini Publik: Pendekatan BERT-based-caused untuk Analisis Sentimen pada Komentar Film," J. Syst. Comput. Eng., vol. 5, no. 1, pp. 36–43, 2024.

[13] N. K. R. Sari, I. M. A. D. Suarjaya, and P. W. Buana, "Perbandingan Translation Library Pada Python (Studi Kasus: Analisis Sentimen Penyakit Menular Di Indonesia)," J. Ilm. Teknol. dan Komput., vol. 2, no. 3, pp. 1–7, 2021.

[14] D. Septiani and I. Isabela, "Analisis Term Frequency Inverse Document Frequency (Tf-Idf) Dalam Temu Kembali Informasi Pada Dokumen Teks," SINTESIA J. Sist. dan Teknol. Inf. Indones., vol. 1, no. 2, pp. 81–88, 2022.

[15] F. Hadaina and U. Budiyanto, "Implementasi Metode Multinomial Naïve Bayes Untuk Sentiment Analysis Terhadap Data Ulasan Produk Colearn Pada Google Play Store," *Semin. Nas. Mhs. Fak. Teknol. Inf. Jakarta-Indonesia*, no. September, pp. 660–666, 2022, [Online]. Available: https://senafti.budiluhur.ac.id/index.php

[16] W. I. Rahayu, C. Prianto, and E. A. Novia, "Perbandingan Algoritma K-Means dan Naive Bayes untuk Memprediksi Prioritas Pembayaran Tagihan Rumah Sakit Berdasarkan Tingkat Kepentingan pada PT. Pertamina (Persero)," J. Tek. Inform., vol. 13, no. 2, pp. 1–8, 2021.