IJCCS (Indonesian Journal of Computing and Cybernetics Systems)
Vol.x, No.x, July xxxx, pp. 1~5
ISSN (print): 1978-1520, ISSN (online): 2460-7258
DOI: 10.22146/ijccs.xxxx

Prediction Sentiment Analysis Grab Reviews Using SVM Linear Based Streamlit

Muhammad Taufiq Hidayat*[1], Muhammad Arifin[2], Syafiul Muzid[3]
[1,2,3]Department of Information System, FT UMK, Kudus, Indonesia
e-mail: ***[1]202153140@std.umk.ac.id**, [2]arifin.m@umk.ac.id, [3]syafiul.muzid@umk.ac.id

Abstrak

Perkembangan teknologi digital telah mempercepat transformasi layanan transportasi daring, yang meningkatkan persaingan dan mendorong inovasi untuk peningkatan kualitas layanan. Sebagai platform unggulan di Indonesia, Grab menghadapi berbagai permasalahan, seperti kualitas pelayanan pengemudi, sistem pembayaran, dan stabilitas aplikasi, yang terefleksikan melalui ulasan pengguna di Google Play Store. Penelitian ini bertujuan memperoleh wawasan strategis melalui evaluasi model Support Vector Machine (SVM) berbasis kernel linear, yang diintegrasikan dalam platform Streamlit, guna memprediksi sentimen ulasan pengguna Grab. Data dikumpulkan melalui web scraping dan diproses dengan teknik tokenisasi, penghapusan stopword, dan stemming untuk meningkatkan akurasi model. Hasil penelitian diimplementasikan dalam situs web interaktif Streamlit dengan dua fitur utama, yaitu prediksi sentimen dan visualisasi plot. Fitur prediksi menampilkan distribusi sentimen, metrik performa, confusion matrix, serta laporan klasifikasi, sedangkan fitur visualisasi menampilkan word cloud, diagram batang, dan diagram lingkaran. Evaluasi model menunjukkan akurasi 83%, presisi 85%, recall 83%, dan F1-score 81% dalam lingkungan Streamlit. Temuan ini diharapkan dapat memberikan kontribusi bagi pengembang dan pemangku kepentingan dalam meningkatkan layanan Grab serta mengembangkan metode prediksi sentimen yang lebih efektif.

Kata kunci*— Prediksi Sentimen, SVM Linear, Streamlit, Grab, Transportasi Online*

Abstract

Advances in digital technology have accelerated the transformation of online transportation services, intensifying competition and driving innovations to enhance service quality. As a leading platform in Indonesia, Grab faces various challenges, including driver service quality, payment systems, and application stability, as reflected in user reviews on Google Play Store. This study aims to gain strategic insights by evaluating a linear kernel-based Support Vector Machine (SVM) model integrated into the Streamlit platform to predict the sentiment of Grab user reviews. Data were collected via web scraping and processed using tokenization, stopword removal, and stemming techniques to improve model accuracy. The model was implemented on an interactive Streamlit website featuring two main functionalities: sentiment prediction and plot visualization. The sentiment prediction feature presents sentiment distribution, performance metrics, a confusion matrix, and a classification report, while the visualization feature displays interactive word clouds, bar charts, and pie charts. The model evaluation in the Streamlit environment demonstrated an accuracy of 83%, a precision of 85%, a recall of 83%, and an F1-score of 81%. These findings are expected to contribute to developers and stakeholders in enhancing Grab services and advancing more effective sentiment prediction methods.

Keywords— *Sentiment Prediction, SVM Linear, Streamlit, Grab, Online Transportation*

Received February 21th, 2025; Revised April 28th, 2025; Accepted July 10th, 2012

1. INTRODUCTION

The rapid advancement of digital technology has revolutionized many industries, notably the app-based transportation sector. Digital platforms have evolved from being mere sources of data to comprehensive ecosystems that offer a wide range of services. This transformation has enabled companies to integrate various functionalities into a single application, thereby enhancing user convenience and operational efficiency [1].

Grab, a leading ride-hailing service in Southeast Asia, has emerged as a pivotal player in this digital transformation. In Indonesia, Grab provides diverse services including transportation, food delivery, logistics, and daily errands. Despite its widespread popularity and extensive driver network covering 224 cities [2], the application faces persistent challenges such as app stability, driver performance, delivery speed, and complex payment systems.

The significance of sentiment analysis in evaluating these challenges cannot be overstated. Sentiment analysis, or opinion mining, automates the extraction of subjective information from user-generated content, thereby providing critical insights into public perception. This analytical approach is essential for understanding user satisfaction and identifying areas in need of service improvement [3].

Previous studies have applied various machine learning techniques to sentiment analysis across diverse contexts. For example, Ar'bah Lailany and Lestari utilized the K-Nearest Neighbors (KNN) method to analyze public sentiment regarding declining marriage rates in Indonesia, despite encountering low precision and recall for negative sentiments [4]. Similarly, Fatkhudin et al. employed Decision Trees to assess perceptions of AI-assisted thesis writing, reporting high recall but low precision for negative sentiments [5]. Pradipta and Widodo further demonstrated robust performance using BERT for social media sentiment analysis during the COVID-19 pandemic, although overfitting was observed in early epochs [6].

Additional research has explored other approaches across digital platforms. Nurhusen et al. applied Logistic Regression to evaluate Twitter sentiments on fuel price hikes, while studies employing LSTM and BI-LSTM reported challenges such as bias toward negative sentiment and high computational demands [7][8][9]. Moreover, Naïve Bayes Classifiers have been used in sentiment analyses for reviews related to McDonald's and Shopee, with limitations including the exclusion of neutral sentiment and low recall for positive reviews [10][11]. Katiandhago et al. and Aryanti et al. further highlighted issues related to low precision in certain contexts, although techniques such as SMOTE have shown promise in improving performance [12][13].

Building on these insights, Support Vector Machine (SVM) has emerged as an effective method for sentiment classification. Comparative studies indicate that SVM often outperforms Naïve Bayes Classifiers in accurately identifying both positive and negative sentiments, making it a strong candidate for handling complex textual data [14][15].

Motivated by the advantages of SVM and the challenges observed in previous research, the current study focuses on predicting sentiment in Grab user reviews. A linear SVM algorithm is employed to classify sentiments from reviews collected on the Google Play Store, aiming to address limitations found in earlier approaches.

Furthermore, this study integrates the SVM model with Streamlit, an open-source framework that enables the development of interactive web applications. Streamlit facilitates the presentation of sentiment analysis results in an accessible and engaging format, thereby enhancing data exploration for both developers and general users [16].

The Grab user-review dataset comprises 6,352 reviews collected between January 1 and November 10, 2024 via Python web scraping, extracting both review text and rating scores. After labeling, 54 % were positive, 39 % negative, and 7 % neutral, revealing user satisfaction trends. The methodology includes data preprocessing (case folding, cleaning, tokenization, stop-word removal, stemming), TF-IDF vectorization, an 80:20 train–test split, linear SVM training with hyperparameter tuning, and performance evaluation (accuracy, precision, recall, F1-score) before deployment in Streamlit. Technically, we fine-tune the SVM's regularization parameter and integrate the pipeline within an interactive interface. Practically, this study enhances the authors' skills in data analysis, contributes to sentiment analysis research, and offers Grab actionable insights for improving services based on real user feedback.

2. METHODS

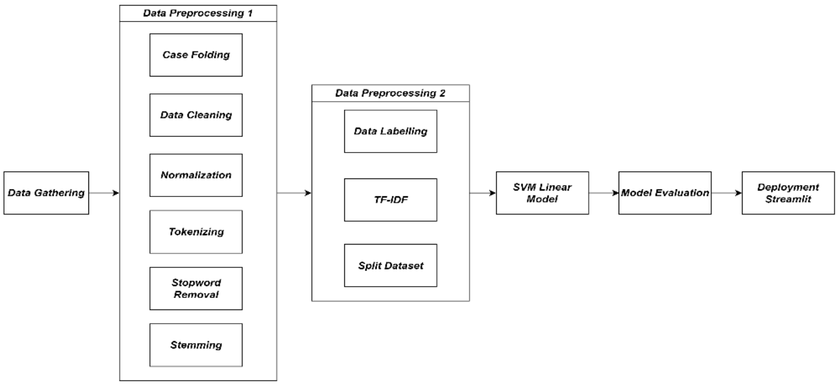


Figure 1 Research method sentiment analysis Grab SVM Linear and Streamlit

Figure 1 above illustrates the methodological framework for analyzing sentiment in Grab user reviews. The process employs the Support Vector Machine Linear (SVM Linear) method, developed using Google Colaboratory, and is ultimately deployed through a website built with Streamlit for user interface. The following is a further explanation of the research flow methodology.

2.1 Data Gathering

In this section, 6,352 user reviews from Grab's Google Play Store (January 1–November 10, 2024) were collected via Python web scraping using the Google Play Scraper. Review text and rating scores were extracted to perform sentiment analysis classifying reviews as positive, neutral, or negative following Deshmane's methodology. [17].

Table 1 Sample data gathering at Google Play Store Grab application

userName	score	at	content
Pengguna Google	5	2024-10-30 08:06:31	"Tolong dong untuk pengelola divisi aplikasinya"
Pengguna Google	5	2024-10-21 12:12:57	"Setelah saya update keversi terbaru, ternyata"
Pengguna Google	3	2024-10-18 12:06:50	"Voucher nya aneh kadang ada kadang juga gaada"
Pengguna Google	1	2024-11-09 10:18:47	"Ada kemungkinan aplikasi nyangkut"
Pengguna Google	2	2024-11-08 06:31:37	"untuk aplikasi ini sebenarnya sangat membantu hanya dalam menyeleksi SDM"

2.2 Data Preprocessing 1

The first stage of data preprocessing was done to filter and prepare the review data for use in further analyses.

2.2.1 Case Folding

In this research, all text is transformed to lowercase (case folding) to standardize the dataset, simplify the vocabulary, remove noise, and prevent misclassification, ultimately improving model performance and ensuring more precise sentiment analysis. The case is illustrated in Table 2 below.

Table 2 Sample case folding process

userName	score	at	content
Pengguna Google	5	2024-10-30 08:06:31	“tolong dong untuk pengelola divisi aplikasinya”
Pengguna Google	5	2024-10-21 12:12:57	“setelah saya update keversi terbaru, ternyata”
Pengguna Google	3	2024-10-18 12:06:50	”voucher nya aneh kadang ada kadang juga gaada”
Pengguna Google	1	2024-11-09 10:18:47	”ada kemungkinan aplikasi nyangkut”
Pengguna Google	2	2024-11-08 06:31:37	”untuk aplikasi ini sebenarnya sangat membantu hanya dalam menyeleksi sdm”

2.2.2 Data Cleaning

In this study, data cleaning eliminates special characters, numbers, and irrelevant symbols from the content column, standardizing the text to minimize noise and enhance the accuracy and reliability of sentiment classification. This process is shown in Table 3 below.

Table 3 Sample data cleaning process

userName	score	at	content
Pengguna Google	5	2024-10-30 08:06:31	“tolong kelola divisi aplikasi”
Pengguna Google	5	2024-10-21 12:12:57	“update versi baru”
Pengguna Google	3	2024-10-18 12:06:50	”voucher nya aneh kadang kadang gaada”
Pengguna Google	1	2024-11-09 10:18:47	”aplikasi nyangkut”
Pengguna Google	2	2024-11-08 06:31:37	”aplikasi bantu seleksi sdm”

2.2.3 Normalization

This stage involves modifying, removing, or correcting words and terms in the content column, including abbreviations, non-standard words, and inconsistent writing variations. Normalization is essential to ensure data uniformity and accuracy, thereby preserving semantic integrity and enhancing the overall quality of subsequent analysis.

2.2.4 Tokenizing

In this section, tokenization segments text into individual tokens (words) from the content column, enabling subsequent normalization, stemming, and stop-word removal. Accurate tokenization is crucial for precise feature extraction and enhances the overall quality of natural language processing tasks.

2.2.5 Stopword Removal

In this process, common words such as "dan" and "yang," which provide no significant information, are removed from the content column through stopword removal. This technique streamlines text analysis by eliminating superfluous lexical items, thereby enhancing data quality and reducing computational complexity.

2.2.6 Stemming

In this section, stemming converts words in the content column to their root forms, standardizing variations and reducing inflectional endings. This normalization enhances data consistency, minimizes textual noise, and improves the overall performance of the sentiment analysis model.

2.3 Data Preprocessing 2

The second stage of data preprocessing involves several essential steps, including data labeling, the application of TF-IDF (Term Frequency-Inverse Document Frequency), and dataset splitting. During this phase, raw text data is converted into numerical feature vectors and divided into training and testing subsets, enabling efficient model training, validation, and performance assessment.

2.3.1 Data Labelling

In this section, dataset entries are labeled into three sentiment classes negative for scores below 3, neutral for a score of 3, and positive for scores above 3 ensuring systematic categorization for accurate pattern recognition in subsequent analysis. This process displayed at Table 4 below this.

Table 4 Sample data labelling process after stemming

userName	score	at	content	label
Pengguna Google	5	2024-10-30 08:06:31	“[’tolong’, ’kelola’, ’divisi’, ’aplikasi’]”	Positif
Pengguna Google	5	2024-10-21 12:12:57	”[’update’, ’versi’, ’baru’]”	Positif
Pengguna Google	3	2024-10-18 12:06:50	“[‘voucher’, ‘nya’, ‘aneh’, ‘kadang’, ‘kadang’, ‘gaada’]”	Netral
Pengguna Google	1	2024-11-09 10:18:47	“[‘aplikasi’, ‘nyangkut’]”	Negatif
Pengguna Google	2	2024-11-08 06:31:37	“[‘aplikasi’, ‘bantu’, ‘seleksi’, ‘sdm’]”	Negatif

This is after preprocessing data using punctuation and lower case on table 5:

Table 5 Sample data labelling process after punctuation and lower case

userName	score	at	content	label
Pengguna Google	5	2024-10-30 08:06:31	”tolong kelola divisi aplikasi”	positif
Pengguna Google	5	2024-10-21 12:12:57	” update versi baru”	positif
Pengguna Google	3	2024-10-18 12:06:50	“voucher nya aneh kadang kadang gaada”	netral
Pengguna Google	1	2024-11-09 10:18:47	“aplikasi nyangkut”	negatif
Pengguna Google	2	2024-11-08 06:31:37	“aplikasi bantu seleksi sdm”	negatif

2.3.2 TF-IDF

Upon completion of the labeling process, we applied the TF-IDF technique to convert textual data into numerical feature vectors. By assessing each term's frequency within a document alongside its inverse frequency across the entire corpus, this method emphasizes the relative importance of words and enables the model to discern vital contextual patterns. Consequently, the "content" column was vectorized to optimize feature representation and strengthen the model's learning capacity.

2.3.3 Split Dataset

During this stage, the dataset is split into 80% for training and 20% for testing to objectively assess the model's generalization capability. The training set enables the model to learn underlying patterns in the data, while the testing set evaluates its performance on previously unseen inputs. This method helps prevent overfitting and promotes consistent model reliability across various data scenarios.

2.4 SVM Linear Model

Based on an article by Elsedimy, the use of Support Vector Machine Linear (SVM Linear) in heart disease risk prediction is an effective approach because SVM Linear is able to linearly separate data between positive and negative classes [18]. On linearly separable data, SVM Linear maximises the margin between data points from the two closest classes to the decision plane. The linear SVM finds the optimal parameters of the weight (w) and bias (b) vectors that determine the decision boundary plane, i.e. $w \cdot x + b = 0$ where the positive and negative classes are separated by hyperplane $H_1$and $H_2$with equations $H_1\colon w \cdot x + b = +1$ and $H_2\colon w \cdot x + b = -1$. This is pattern of SVM Linear on Equation (1):

$$Minimize = \frac{1}{2}\|\vec{w}\|^2$$
$$y_i(\vec{w}^T.\vec{x}_i + b) \geq 1, \forall_i = 1,2,\dots,N \quad (1)$$

In equation (1), some of the main components that define the linear SVM classification process are as follows. The weight vector $(\vec{w})$ determines the direction and orientation of the hyperplane that distinguishes the positive and negative classes. The value of w is controlled to minimise $\|\vec{w}\|^2$, with the aim of maximising the margin or distance between the data points of the two classes with respect to the hyperplane. Bias (b) is a constant that adjusts the position of the hyperplane so that the separation between positive and negative classes is in accordance with the margin constraint. The margin constraint $y_i(\vec{w}^T.\vec{x}_i + b) \geq 1$ ensures that each data point from both classes has minimal margin to the hyperplane. When this constraint is satisfied for all data points iii, the classification results show that there is a maximum distance separating the two classes.

2.5 Model Evaluation

In this section, we assess the linear SVM's performance using accuracy, precision, recall, and F1-score. These metrics validate the model's ability to correctly classify user sentiments and inform data-driven enhancements to the service.

2.5.1 Accuracy

Accuracy (A) serves as a fundamental metric for assessing a model's ability to correctly classify data [19]. It plays a vital role in sentiment analysis by ensuring precise categorization of user opinions. A high accuracy score indicates the model's effectiveness in distinguishing sentiment classes, thereby providing valuable insights for informed decision-making. The calculation of accuracy follows the formula presented in Equation (2):

$$A = \frac{TP+TN}{Total\ samples} \quad (2)$$

2.5.2 Precision

Precision (P) measures how accurately a model identifies the correct sentiment among the predicted positive instances [19]. It helps minimize false positives, ensuring sentiment classifications remain relevant. High precision values indicate the model's ability to accurately identify sentiment, which is vital for applications where incorrect classifications can impact decisions. The precision formula is presented in Equation (3):

$$P = \frac{TP}{TP+FP} \tag{3}$$

2.5.3 Recall

Recall (R) quantifies a model's capability to correctly identify and classify sentiments within a given dataset [19]. A higher recall score signifies the model's proficiency in detecting relevant instances while reducing the occurrence of false negatives. The calculation for recall is presented in Eq. (4) below:

$$R = \frac{TP}{TP+FN} \tag{4}$$

2.5.4 F1-Score

F1-Score (F) is a harmonic mean of precision and recall, providing a balanced evaluation of a model's overall performance [19]. It is especially useful for imbalanced datasets, preventing misleading evaluations based on accuracy alone. A high F1-score ensures optimized precision and recall for robust sentiment classification. The F1-score formula is presented in Equation (5):

$$F = 2 \times \frac{P \times R}{P+R} \tag{5}$$

2 6 Deployment Streamlit

The Streamlit-based deployment model delivers sentiment analysis results in an interactive format, enhancing accessibility for both developers and general users [16]. The linear SVM model is integrated into the Streamlit platform, providing two primary features: a sentiment prediction interface and a data visualization dashboard. The prediction interface enables users to input review text and receive sentiment labels along with confidence scores, a confusion matrix, and a detailed classification report. Meanwhile, the visualization dashboard includes exploratory data analysis (EDA) tools such as word clouds, bar charts for review ratings, pie charts illustrating sentiment distribution, and interactive confusion matrix heatmaps. These features allow developers and stakeholders to efficiently explore sentiment patterns and extract meaningful, data-driven insights.

3. RESULTS AND DISCUSSION

This section presents the research findings and analysis of the sentiment analysis model for Grab user reviews. It highlights key libraries used in the Streamlit environment, including Streamlit, Pandas, Matplotlib, Seaborn, WordCloud, and Scikit-learn for data handling, visualization, text vectorization, modeling, evaluation, and optimization, ensuring a streamlined and efficient workflow from data collection to sentiment classification.

Table 6 Requirement Streamlit

Library	**Function**
streamlit	Build interactive web apps
streamlit_option_menu: option_menu	Create navigation menu
os	Access operating system functions
pandas	Data manipulation and analysis
matplotlib.pyplot	Plot graphs and visualizations

seaborn	Advanced statistical visualization
wordcloud: WordCloud	Generate word cloud images
pickle	Serialize Python objects
sklearn.feature_extraction.text: TfidfVectorizer	Convert text to TF-IDF features
sklearn.model_selection: train_test_split	Split dataset for training and testing
joblib	Save and load model objects
sklearn.svm: SVC	Support vector classification model
sklearn.metrics: accuracy_score, classification_report, confusion_matrix, precision_score, recall_score, f1_score	Evaluate model performance metrics
sklearn.model_selection: GridSearchCV	Hyperparameter tuning for SVM

The Streamlit model deployment is divided into two main sections: Sentiment Prediction and Plot Visualization. The Sentiment Prediction section delivers detailed results with interactive visuals, including sentiment distribution, model performance metrics, a confusion matrix, and classification reports. Meanwhile, the Plot Visualization section features interactive displays such as word clouds, bar charts, and pie charts, providing valuable insights that enhance the accessibility of sentiment analysis and support better decision-making for improving the Grab application.

3.1 Sentiment Prediction

The Sentiment Prediction page uses a linear SVM model to analyze Grab service reviews from the Play Store. Hyperparameter tuning via GridSearchCV optimizes the C parameter for improved accuracy. The application features sections for sentiment prediction, model evaluation metrics, a confusion matrix, and sentiment results on the original dataset, providing comprehensive insights into classification performance and supporting data-driven enhancements in online transportation services.

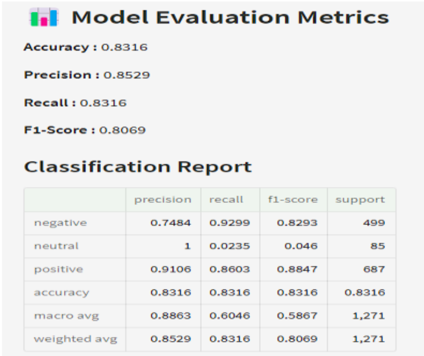
3.1.1 Review Sentiment Prediction

In this section, users can enter their review into a text input field, and the system will immediately predict the sentiment along with the model's confidence score. The prediction output is presented in a clear and user-friendly format to facilitate easy interpretation. The analysis begins by converting the user input into a TF-IDF vector, which is then processed by the trained Support Vector Machine (SVM) linear model to determine the sentiment based on learned patterns. If no input is detected, the system prompts the user to provide a review, ensuring usability and meaningful engagement. This real-time prediction feature supports both users and developers in gaining insight into user sentiment. The model categorizes input into three sentiment classes: positive, neutral, and negative, as illustrated in Figure 2 below.

Figure 2 Review sentiment prediction page is positive, negative, and neutral

3.1.2 Model Evaluation Metrics

Beyond the detailed classification report and confusion matrix, we evaluated the model using accuracy, precision, recall, and F1-score metrics. These measures provide an objective and transparent assessment of sentiment classification performance across all categories. The results are presented in Figure 3 below:


Model Evaluation Metrics

Accuracy : 0.8316

Precision : 0.8529

Recall : 0.8316

F1-Score : 0.8069

Classification Report

	precision	recall	f1-score	support
negative	0.7484	0.9299	0.8293	499
neutral	1	0.0235	0.046	85
positive	0.9106	0.8603	0.8847	687
accuracy	0.8316	0.8316	0.8316	0.8316
macro avg	0.8863	0.6046	0.5867	1,271
weighted avg	0.8529	0.8316	0.8069	1,271

Figure 3 Sentiment prediction metrics evaluation model page with Streamlit

Table 7 Comparison of Streamlit evaluation metrics model with Google Colab

Metrics	Streamlit	Google Colaboratory
Accuracy	83%	84%
Precision	85%	79%
Recall	83%	84%
F1-Score	81%	81%

Table 7 indicates that the Streamlit-based model achieved an accuracy of 83%, compared to 84% on Google Colaboratory. Precision was higher on Streamlit (85% vs. 79%), though its recall was slightly lower (83% vs. 84%). The F1-score was consistent at 81% for both. These results underscore the trade-offs between deployment environments and emphasize the need for further optimization to enhance recall performance while maintaining high precision, reliability, and overall robustness of the model.

3.1.3 Confusion Matrix

The Confusion Matrix evaluates SVM sentiment classification by comparing actual and predicted labels, detailing accuracy, precision, recall, and F1-score in a heatmap. Its interactive Streamlit implementation highlights misclassifications to refine model reliability. Figure 4 illustrates these performance metrics.

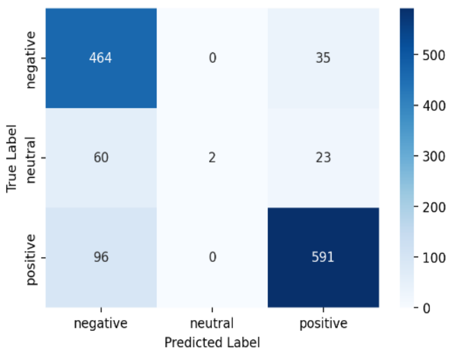


Figure 4 Confusion matrix with Streamlit

3.1.4 Sentiment Prediction Result Label

A trained SVM model generated sentiment predictions for Grab app reviews collected from the Google Play Store. After applying TF-IDF vectorization, predicted labels were assessed against actual labels. The results were then presented through an interactive interface showcasing ten sample reviews and exported as a CSV file for further analysis (see Figure 5).

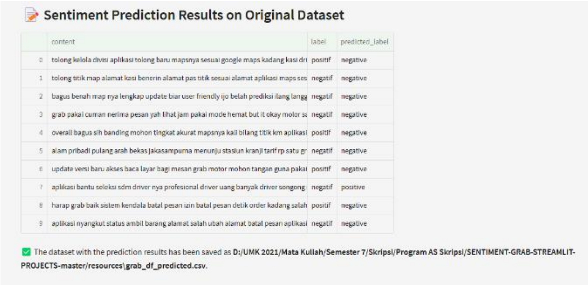


Figure 5 Label sentiment prediction page with Streamlit

3.2 Plot Visualization

The Plot Visualization module functions as the primary interface for exploring user sentiment in Grab app reviews, incorporating features such as review score distribution, sentiment proportion charts, and word cloud analysis. These dynamic, interactive graphics improve accessibility and yield crucial insights into overall user satisfaction and the most frequently discussed themes in the feedback.

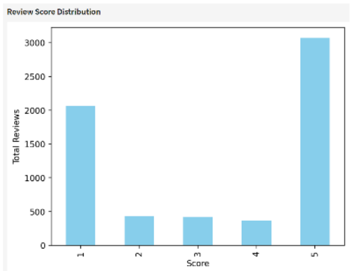


Figure 6 Bar Chart Review Score Distribution

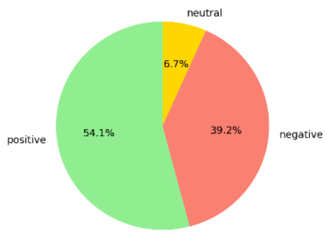


Figure 7 Pie chart sentiment propotion analysis using Streamlit

Figure 8 Wordcloud sentiment positive, neutral, and negative

Figure 6 illustrates the distribution of Grab's review scores in a bar chart, showing that about 3,000 users awarded the app five stars, while fewer than 500 rated it four stars—evidence of high overall satisfaction and hints at areas for improvement. Figure 7's pie chart breaks down user sentiment into 54 % positive (light green), 39 % negative (salmon), and 7 % neutral (gold), providing clear guidance for targeted service enhancements. Lastly, Figure 8 displays word clouds by sentiment: the positive cluster highlights words like "ramah layananya," "grab," "bantu," "aplikasi," and "cepat," whereas the neutral and negative clusters feature terms such as "kadang," "titik," "driver," "pesannya," "mohon," "susah," "kecewa," "harga," "order," and "makan."

4. CONCLUSIONS

In conclusion, the proposed sentiment analysis pipeline was successfully deployed via Streamlit, offering an intuitive interface for both sentiment prediction and plot visualization. The implementation achieved an accuracy of 83%, precision of 85%, recall of 83%, and an F1-score of 81%. In comparison, the Google Colaboratory environment yielded a slightly higher accuracy (84%) but lower precision (79%). The real-time sentiment prediction feature reliably classified reviews into positive, neutral, and negative categories, while performance evaluation—including accuracy, precision, recall, F1-score, and the confusion matrix—confirmed the model's robustness and pinpointed areas for improvement. Plot visualization tools such as bar charts, pie charts, and word clouds enriched insights: the bar chart highlighted roughly 3,000 five-star ratings versus fewer than 500 four-star ratings, the pie chart showed 54% positive, 39% negative, and 7% neutral sentiments, and the word clouds revealed distinct lexical patterns across sentiment classes. Together, these plot visualizations offer a comprehensive understanding of user sentiment and highlight potential areas for service improvement.

Despite these successes, the study identified limitations, notably a slight drop in accuracy and recall in the Streamlit deployment and occasional misclassifications. These issues underscore the need for further hyperparameter tuning and the adoption of more advanced natural language processing techniques. Future work should explore alternative classification algorithms, refine preprocessing strategies, and integrate state-of-the-art models to boost the overall reliability and effectiveness of sentiment prediction for the Grab application.

REFERENCES

[1] A. Yosediputra and E. Supriyono, "EFFECT OF PERCEIVED USEFULNESS AND PERCEIVED EASE OF USE TO USAGE DECISION GRAB ONLINE TRANSPORTATION SERVICE IN SIDOARJO REGENCY AREA," *International Journal of Economy, Education, and Entrepreneurship*, vol. 4, no. 2, pp. 408–418, 2024, doi: 10.53067/ije3.v4i2.

[2] A. Zahira Haerul, "SEIKO : Journal of Management & Business Pengaruh User Interface, Price Discount, Reputasi Perusahaan, dan Kemudahan Penggunaan terhadap Keputusan Pembelian Melalui Aplikasi Grab oleh Konsumen Millennials Di Kota Makassar," *SEIKO : Journal of Management & Business*, vol. 7, no. 1, pp. 652–664, 2024.

[3] I. Sugiyarto, S. Anggraeni, U. Faddilah, and A. A. Muzaffar, "Sentimen Analisis Pengguna Aplikasi Grab Menggunakan Algoritma Naive Bayes Classifier dan Support Vector Machine," *JURNAL TEKNIKA*, vol. 18, no. 1, pp. 331–341, 2024.

[4] A. Ar'bah Lailany and S. Lestari, "Analisis Sentimen Publik Terhadap Penurunan Jumlah Pernikahan di Indonesia menggunakan Algoritma K-Nearest Neighbors (KNN)," *Jurnal Indonesia : Manajemen Informatika dan Komunikasi (JIMIK)*, vol. 5, no. 3, pp. 3043–3053, 2024, [Online]. Available: https://journal.stmiki.ac.id

[5] A. Fatkhudin, F. Adi Artanto, N. Abiyu Safli, and D. Wibowo, "Decision Tree Berbasis SMOTE dalam Analisis Sentimen Penggunaan Artificial Intelligence untuk Skripsi," *Remik: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 8, no. 2, pp. 494–505, 2024, doi: 10.33395/remik.v8i2.13531.

[6] D. Pradipta and E. Widodo, "Sentiment Analysis on Social Media using Bidirectional Encoder from Transformers (Case Study : Covid-19 Omicron)," *INFORMASI (Jurnal Informatika dan Sistem Informasi)*, vol. 16, no. 2, pp. 267–281, 2024.

[7] M. R. Nurhusen, J. Indra, and K. A. Baihaqi, “Analisis Sentimen Pengguna Twitter Terhadap Kenaikan Harga Bahan Bakar Minyak (BBM) Menggunakan Metode Logistic Regression,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 7, no. 1, pp. 276–282, Jan. 2023, doi: 10.30865/mib.v7i1.5491.

[8] E. Setyaningtyas and K. Nugroho, “Analisis Sentimen Media Sosial Pada Pengguna Twitter Terhadap Pemilu 2024 Menggunakan Metode LSTM,” *Jurnal Riset Sistem Informasi Dan Teknik Informatika (JURASIK)*, vol. 9, no. 2, pp. 673–683, 2024, [Online]. Available: https://tunasbangsa.ac.id/ejurnal/index.php/jurasik

[9] R. Onsu, D. F. Sengkey, and F. D. Kambey, “IMPLEMENTASI BI-LSTM DENGAN EKSTRAKSI FITUR WORD2VEC UNTUK PENGEMBANGAN ANALISIS SENTIMEN APLIKASI IDENTITAS KEPENDUDUKAN DIGITAL,” *Jurnal Teknologi Terpadu*, vol. 10, no. 1, pp. 46–55, 2024.

[10] S. D. S. Kurniawan and A. Fauzy, “Penggunaan Naïve Bayes Classifier dalam Analisis Sentimen Ulasan Aplikasi McDonald’s: Perspektif Pengguna di Indonesia,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 3, p. 1545, Jul. 2024, doi: 10.30865/mib.v8i3.7765.

[11] K. M. Elistiana, Bagus Adhi Kusuma, P. Subarkah, and H. A. Awal Rozaq, “IMPROVEMENT OF NAIVE BAYES ALGORITHM IN SENTIMENT ANALYSIS OF SHOPEE APPLICATION REVIEWS ON GOOGLE PLAY STORE,” *Jurnal Teknik Informatika (JUTIF)*, vol. 4, no. 6, pp. 1431–1436, Dec. 2023, doi: 10.52436/1.jutif.2023.4.6.1486.

[12] B. J. Katiandhago, A. Mustolih, W. D. Susanto, P. Subarkah, and C. I. Satrio Nugroho, “Sentiment Analysis of Twitter Cases of Riots at Kanjuruhan Stadium Using the Naive Bayes Method,” *Journal of Computer Networks, Architecture and High Performance Computing*, vol. 5, no. 1, pp. 302–312, Apr. 2023, doi: 10.47709/cnahpc.v5i1.2196.

[13] R. Aryanti, T. Misriati, and A. Sagiyanto, “Analisis Sentimen Aplikasi Primaku Menggunakan Algoritma Random Forest dan SMOTE untuk Mengatasi Ketidakseimbangan Data,” *Journal of Computer System and Informatics (JoSYC)*, vol. 5, no. 1, pp. 218–227, Nov. 2023, doi: 10.47065/josyc.v5i1.4562.

[14] Sarimole. Frencis Matheos and Kudrat, “Analisis Sentimen Terhadap Aplikasi Satu Sehat Pada Twitter Menggunakan Algoritma Naive Bayes Dan Support Vector Machine,” *Jurnal Sains dan Teknologi*, vol. 5, no. 3, pp. 783–790, 2024, doi: 10.55338/saintek.v5i1.2702.

[15] W. Ningsih, B. Alfianda, R. Rahmaddeni, and D. Wulandari, “Perbandingan Algoritma SVM dan Naïve Bayes dalam Analisis Sentimen Twitter pada Penggunaan Mobil Listrik di Indonesia,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 2, pp. 556–562, Feb. 2024, doi: 10.57152/malcom.v4i2.1253.

[16] D. Leni, A. Karudin, M. R. Abbas, J. K. Sharma, and A. Adriansyah, “OPTIMIZING STAINLESS STEEL TENSILE STRENGTH ANALYSIS: THROUGH DATA EXPLORATION AND MACHINE LEARNING DESIGN WITH STREAMLIT,” *EUREKA, Physics and Engineering*, vol. 2024-September, no. 5, pp. 73–88, Sep. 2024, doi: 10.21303/2461-4262.2024.003296.

[17] V. Deshmane, J. Musale, S. Joshi, V. Chinta, K. Gokak, and I. Dalbhanjan, “Web Scraping for E-Commerce Website To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code,” *International Journal for Innovative Engineering and Management Research*, vol. 13, no. 4, pp. 216–224, 2024, doi: 10.48047/IJIEMR/V13/ISSUE.

[18] E. I. Elsedimy, S. M. M. AboHashish, and F. Algarni, “New cardiovascular disease prediction approach using support vector machine and quantum-behaved particle swarm optimization,” *Multimed Tools Appl*, vol. 83, no. 8, pp. 23901–23928, Mar. 2024, doi: 10.1007/s11042-023-16194-z.

[19] B. Ramadhani and R. R. Suryono, “Komparasi Algoritma Naïve Bayes dan Logistic Regression Untuk Analisis Sentimen Metaverse,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 2, p. 714, Apr. 2024, doi: 10.30865/mib.v8i2.7458.