Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

Analyzing Sentiment Of I.Saku E-Wallet Play Store Reviews Using Support Vector Machine

Sidik Praptomo[1], Suryanto[2]
[1]Universitas Muhammadiyah Muara Bungo, [2]STIKes Mitra Husada Kaangannyar
e-mail: [1]sidikpraptomo7@gmail.com, [2]suryantosuryan@gmail.com

Abstract

The digital era has transformed lifestyles, with people increasingly relying on electronic devices as tools for everyday life. Technology simplifies various activities by streamlining data and information processing. The growing demand for data has driven the development of new technologies to process information quickly. Technological advancements have improved transportation, access to information, education, and the convenience of online transactions, particularly through digital wallets or e-wallets. Digital financial services, including e-wallets, facilitate easy transactions using e-wallet funds. The primary reason for their use is the convenience of electronic wallets, which eliminate the need for cash and simplify transactions for both buyers and sellers. The case of i.Saku demonstrates its popularity, with over 5 million downloads on the Google Play Store since 2017. Google's digital platform, Google Play Store, includes user reviews as a valid source of information. Nearly 50% of internet users rely on recommendations from other users before using a product. Google Play Store reviews influence the decisions of potential users, but managing them manually is not easy. Sentiment analysis, or opinion mining, is the study of opinions, behaviors, and feelings of individuals towards an entity and is crucial for understanding user reviews. In the context of i.Saku, this research focuses on sentiment analysis of Google Play Store reviews using support vector machine techniques. The study outlines the stages from preprocessing to sentiment analysis, highlighting the complexity and benefits of technology-based sentiment analysis.

Keywords Sentiment Analysis Support Vector Machine Google Play Store i.Saku E-Wallet.

Keywords: *Sentiment Analysis Support Vector Machine Google Play Store i.Saku E-Wallet*

INTRODUCTION

The rapid evolution of financial technology (fintech) has driven the widespread adoption of electronic wallets (e-wallets), enabling users to conduct transactions quickly and securely through mobile applications. In Indonesia, one notable e-wallet platform is i.Saku, a digital payment service developed by Indomaret that offers seamless retail integration and user convenience [1], [4].

With the increasing reliance on digital financial services, user feedback available on platforms such as the Google Play Store has become an essential source of information for evaluating user satisfaction, identifying usability issues, and guiding service improvements. However, the unstructured nature and vast volume of user reviews necessitate automated analysis techniques to derive actionable insights [1], [2].

Sentiment analysis, a key area within natural language processing (NLP), is used to extract subjective opinions from textual data, determining whether user sentiments are positive, negative, or neutral. This analysis provides a deeper understanding of public perception and user experience [2], [3]. Among the various classification algorithms used in sentiment analysis, Support Vector Machine (SVM) is widely recognized for its high performance in processing high-dimensional text data [3], [5].

Recent research has demonstrated the effectiveness of combining SVM with feature extraction methods such as Term Frequency-Inverse Document Frequency (TF-IDF) to improve sentiment classification performance in analyzing mobile app reviews [3], [4]. In this study, we apply sentiment analysis techniques to Play Store reviews of the i.Saku app using the SVM model. The goal is to classify sentiments

Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

and uncover trends in user opinions regarding app performance, usability, and customer experience.

This study includes the following stages: data collection from the Play Store, preprocessing of review texts, TF-IDF-based feature extraction, and sentiment classification using SVM. The insights obtained are expected to be beneficial for developers and stakeholders to better understand user expectations and enhance service quality [1], [5].

RESEARCH METHOD

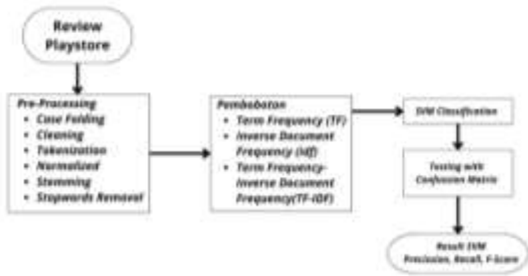

Figure 1. Research Metodology Flow

In Figure 1, the research methodology illustrates the process of the entire study conducted and consists of the following elements:

2.1 Review Playstore

This section explains where and how the data was obtained, specifically by searching for and examining the available data. The relevant data was then collected, followed by a selection process to ensure that the data truly met the research needs [6].

2.2 Preprocesing

Preprocessing is the initial processing step for selecting words from text data in Google Play Store reviews. This involves choosing and removing unnecessary words to obtain more concise user emotion terms. Therefore, to process the reviews, it is necessary to remove certain elements from the review text. The preprocessing steps include case folding, cleansing, tokenizing, normalization, stemming, and stopword removal [7].

2.3 Feature Extraction (TF-IDF)

In this stage, each document undergoes weighting or scoring calculations using a dictionary such as KBBI (the Indonesian Dictionary) or a slang dictionary, which contains scores created for matching with words in the document [15]. The words are then represented in vector form and TF-IDF, and an information table is created that includes Term Frequency (TF), Document Frequency (DF), and IDF for each term, followed by multiplying TF and IDF [4]. The goal is to obtain labels/sentiments for each term/word in the document [8].The IDF formula used is: $IDF(t) = \log(N / df(t))$, where N is the number of documents and df(t) is the number of documents containing term t.

2.4 SVM Clasification

SVM classification is a machine learning technique used for classification and regression tasks. Its focus is on classification tasks, where SVM aims to build a model that can separate two classes of data in a high-dimensional space. The goal is to find the optimal hyperplane that divides the two classes. SVM classification [9] can be applied to various types of data, whether two-dimensional or high-dimensional. One advantage of SVM is its ability to naturally handle data that is not linearly separable using kernel functions. Kernels allow data to be mapped into a higher-dimensional space where it can be more easily separated by hyperplanes [14]. As part of the aforementioned sentiment analysis, SVM can be used to classify reviews or texts into positive or negative sentiment categories. By learning from labeled training data, SVM can identify patterns and build a model that can be used to predict the sentiment of new text [10].

2.5 Accuration Testing with Confussion Matrix

Accuracy testing with a confusion matrix is an evaluation method used in machine learning to measure how well a classification model predicts correctly[13]. The confusion matrix provides a detailed view of the model's performance by comparing the true and false predictions, helping to identify true positives, false positives, true negatives, and false negatives [11]. By using this information, we can calculate various evaluation metrics such as accuracy, precision, recall, and F1

Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

score, which offer deeper insights into the model's effectiveness in handling different classes or categories.

2.6 Result

Section six of this research discusses the results of the sentiment analysis, providing a comprehensive overview of the responses and feedback received from the collected data. This analysis illustrates the feelings and opinions emerging from various sources, offering valuable insights into the sentiments related to the researched topic.

RESULT AND ANALYSIS

The discussion of the classification process using the SVM method will be explained in the following section.

3.1. Review Playstore Classification

The crawling process resulted in 1,000 review data from the Google Play Store for the i.Saku e-wallet application by utilizing the features on the website "https://exportcomments.com/." This method facilitates the crawling process, as there are no limitations on the number of reviews, and the results can be directly saved as a ".csv" file and then stored as an Excel file. An example of a document resulting from the crawling process can be seen in Table 1.

Table 1. Example Data Crawling

No.	Term
Doc 1	Terima kasih dan sangat membantu
Doc 2	Gagal login lupa password malah yang ada member tidak ditemukan
Doc 3	Sangst mudah dan membantu sekali gak ribet (sat set)

3.2. Processing

3.2.1 Case Folding and Cleansing

ISSN 1978-1660
ISSN *online* 2549-8517

This process involves standardizing uppercase letters into lowercase and cleaning the document of unnecessary words to reduce noise. The words removed include HTML characters, keywords, emoticons, hashtags (#), username mentions (@username), numbers, and URLs. An example taken from the first document can be seen in Table 2.

Tabel 2. Case Folding Dan Cleansing

No.	*Term*
Doc 1	Terimakasih dan sangat membantu
Doc 2	gagal login lupa password malah yang ada member tidak ditemukan
Doc 3	sangst mudah dan membantu sekali gak ribet sat set

3.2.2 Tokenizing

This process involves selecting, splitting, and segmenting words in the document into terms based on spaces. An example of the tokenization results can be seen in Table 3.

Table 3. Tokenizaton

No.	Term
Doc 1	[‘terimakasih‘, ‘dan‘, ‘sangat‘, ‘membantu‘]
Doc 2	[‘gagal‘, ‘login‘, ‘lupa‘, ‘password‘, ‘malah‘, ‘yang‘, ‘ada‘, ‘member‘, ‘tidak‘, ‘ditemukan‘]
Doc 3	[‘sangst‘, ‘mudah‘, ‘dan‘, ‘membantu‘, ‘sekali‘, ‘gak‘, ‘ribet‘, ‘sat‘, ‘set‘]

3.2.3 Normalization

This stage converts informal/slang words into standard words according to the KBBI, as shown in the example in Table 4.

Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

Tabel 4. Normalization

No.	Term
Doc 1	[‘terimakasih‘, ‘dan‘, ‘sangat‘, ‘membantu‘]
Doc 2	[‘gagal‘, ‘login‘, ‘lupa‘, ‘password‘, ‘malah‘, ‘yang‘, ‘ada‘, ‘member‘, ‘tidak‘, ‘ditemukan‘]
Doc 3	[‘sangat‘, ‘mudah‘, ‘dan‘, ‘membantu‘, ‘sekali‘, ‘tidak‘, ‘ribet‘]

3.2.4 Stemming

This process involves removing prefixes or suffixes from words that include conjunctions, prepositions, or pronouns, reducing them to their root form according to the KBBI. The results of the stemming process are presented in Table 5.

Tabel 5. Stemming

No.	Term
Doc 1	[‘terimakasih‘, ‘dan‘, ‘sangat‘, ‘bantu‘]
Doc 2	[‘gagal‘, ‘login‘, ‘lupa‘, ‘password‘, ‘malah‘, ‘yang‘, ‘ada‘, ‘member‘, ‘tidak‘, ‘temu‘]
Doc 3	[‘sangat‘, ‘mudah‘, ‘dan‘, ‘bantu‘, ‘sekali‘, ‘tidak‘, ‘ribet‘]

3.2.5 Stopwords Removal

This is the process of filtering out words in the document that are irrelevant to sentiment analysis, as presented in Table 6.

Tabel 6. Stopwords Removal

No.	Term
Doc 1	[‘terimakasih‘, ‘bantu‘]
Doc 2	[‘gagal‘, ‘login‘, ‘lupa‘, ‘password‘,‘member‘, ‘tidak‘, ‘temu‘]

Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

Doc 3	‘mudah‘, ‘bantu‘, ‘sekali‘, ‘tidak‘, ‘ribet‘]

3.3 Class Weghting

Table 7. Classweighting

Term	TF			DF	IDF	TF x IDF		
	D1	D2	D3			D1	D2	D3
terimakasih	1	0	0	1	0,47	0,47	0	0
bantu	1	0	0	1	0,47	0,47	0	0
gagal	0	1	0	1	0,47	0	0,47	0
login	0	1	0	1	0,47	0	0,47	0
lupa	0	1	0	1	0,47	0	0,47	0
member	0	1	0	1	0,47	0	0,47	0
password	0	1	0	1	0,47	0	0,47	0
tidak	0	1	0	1	0,47	0	0,47	0
temu	0	1	0	1	0,47	0	0,47	0
mudah	0	0	1	1	0,47	0	0	0,47
sekali	0	0	1	1	0,47	0	0	0,47
ribet	0	0	1	1	0,47	0	0	0,47

From Table 7, the TF-IDF results for the documents used as test data in this study were obtained. The TF column indicates how frequently a word/term appears in the test document. The IDF column is derived from [10]; for example, the term “terimakasih” yields a TF-IDF result by multiplying TF by IDF: 1 * 0.477 = 0.477.

3.4 SVM Classification

The following manual calculation of the decision function f(x) is presented only as an illustrative example to explain how SVM works mathematically. The values of α_i, y_i, and bias (b) shown in this example do not originate from the real model training results. The actual SVM classification in this study uses the scikit-learn

ISSN 1978-1660
ISSN *online* 2549-8517

library, where the support vectors, coefficients, and bias are computed internally by the algorithm. Therefore, the numerical values from the real model are not manually presented in this section.

In the sentence "Terima kasih dan sangat membantu," after undergoing the preprocessing process, it becomes ["terimakasih", "membantu"]. These terms are converted into the matrix K(x,x1), which represents the frequency of word occurrence in the test sample across 12 terms. The matrix K(x,x1) can be seen in Table 8.

Table 8. Matrix K(X,X1)

	X1	X2
K(x,x1)	0,477	0,477

From Table 8, the results for obtaining the values of x,x1 are shown in Table 9.

Tabel 9. Matrix X,X1

0,477	0,477	0,477+0,477=0,954
1	1	

Table 9 shows the results for each term's weight, calculated as follows:

f(x)=a1y1k(x,x1)+a2y2k(x,x2)+b

f(x)=(1∗(1)∗0.412125)+(1∗(1)∗0.412125)+1=1.942425f(x) = (1 * (1) * 0.412125) + (1 * (1) * 0.412125) + 1 = 1.942425

Thus, f(x) sign(1.942425)=1

Therefore, the new sentence "Terima kasih dan sangat membantu" is classified into class (1) or positive class.

3.5 Confussion Matrix

Testing is performed using a confusion matrix via the SVM library, based on the training and testing models that have been previously classified. This results in a 2x2 matrix representing the actual classes and predicted classes. The outcome of the training model is evaluated with new, previously untrained data. The confusion matrix is presented in Table 10.

Table 10. Confusion Matrix

Actual Data	Predicyion	
	Positive	Negative
Positive	TP : 8	FP : 9
Negative	FN : 3	TN : 33

In the confusion matrix, TP, FP, TN, and FN are abbreviations for different types of prediction outcomes in the context of classification:

True Positive (TP): The number of positive observations that were correctly predicted by the model. This means the model correctly identifies positive examples.

False Positive (FP): The number of negative observations incorrectly predicted as positive by the model. This means the model gave a positive prediction when it should have been negative.

True Negative (TN): The number of negative observations that were correctly predicted by the model. This means the model correctly identifies negative examples.

False Negative (FN): The number of positive observations incorrectly predicted as negative by the model. This means the model gave a negative prediction when it should have been positive [12]. The accuracy of the SVM method can be calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} = \frac{41}{53} = 0{,}77$$

The confusion matrix testing is implemented in Python code to generate an accuracy report, as explained in Figure 2.

Jurnal Dinamika Informatika
Volume 14, No 2, Oktober 2025
ISSN 1978-1660
ISSN *online* 2549-8517

```python
# Membuat classification report dengan imbalanced data
from sklearn.metrics import classification_report


## Term presence ##
print('Imbalanced data - Term presence\n',
      classification_report(Test_Y_Actual, predictions_SVM_TP))

## TF-IDF ##
print('Imbalanced data - TF-IDF\n',
      classification_report(Test_Y_Actual, predictions_SVM_Tfidf))
```

Fig. 2 Python Source Code

3.5 SVW Abalyze

The results from testing with 1,000 data samples on Google Play Store reviews yielded accuracy, precision, recall, and F-1 Score values, which can be seen in Figure 3.

```
Imbalanced data - Term presence
              precision    recall  f1-score   support

           0       0.79      0.86      0.83        36
           1       0.64      0.53      0.58        17

    accuracy                           0.75        53
   macro avg       0.72      0.70      0.70        53
weighted avg       0.75      0.75      0.75        53

Imbalanced data - TF-IDF
              precision    recall  f1-score   support

           0       0.79      0.92      0.85        36
           1       0.73      0.47      0.57        17

    accuracy                           0.77        53
   macro avg       0.76      0.69      0.71        53
weighted avg       0.77      0.77      0.76        53
```

Fig. 3. SVM Testing Results

The testing results yielded values for accuracy, precision, recall, and F-1 Score, which are presented in Figure 3. The SVM method, using scikit-learn's TF-IDF calculation, achieved an accuracy score of 0.77. SVM demonstrated higher accuracy compared to the Term-presence method, which only achieved an accuracy score of 0.75 in the tests.

CONCLUSION

This study aims to evaluate the performance of the Support Vector Machine (SVM) method in sentiment analysis using a dataset of 1,000 samples. The data processing revealed an imbalanced distribution, with 65.52% of the data identified as negative sentiment and the remaining 34.48% as positive sentiment. This analysis provides an initial overview of the dataset characteristics, which is crucial for evaluating the performance of the SVM model.

In the context of sentiment analysis, evaluating the model's performance is critical for understanding how well it can classify sentiment with good accuracy. The performance metrics used in this test include precision, recall, F-1 score, and overall accuracy. The results show that the SVM model achieved an accuracy rate of 77%, which is considered a significant accomplishment.

Furthermore, a comparison was made with the Term Presence method. In this comparison, SVM outperformed with a higher accuracy. It was found that Term Presence only achieved an accuracy of 75%. This result highlights the advantage of SVM in handling sentiment analysis tasks on the dataset used.

Suggested future research: subsequent studies are recommended to address data imbalance using methods such as SMOTE or undersampling, perform hyperparameter tuning on SVM, compare different SVM kernels, and evaluate the performance of deep learning algorithms as benchmarks.

REFERENCES

[1] I. R. Putri and R. F. Sari, "Sentiment analysis on customer reviews of mobile payment applications using naïve Bayes and support vector machine," Procedia Computer Science, vol. 179, pp. 375–382, 2020.

[2] R. Alharbi and S. Khan, "Sentiment analysis of Arabic tweets: A survey," IEEE Access, vol. 9, pp. 110389–110416, 2021.

[3] M. A. Alomari, J. N. Al-Karaki, and Z. AlShara, "A hybrid approach of SVM and deep learning for sentiment classification in social media," Applied Sciences, vol. 13, no. 5, p. 2782, 2023.

[4] C. P. Dellavianti Nishfi Ilmiah Huda, "Analisis Sentimen Layanan Jasa Pengiriman Pada Ulasan Play Store: Systematic Literature Review," Jurnal Informatika dan Teknologi Komputer, vol. 04, no. UNSAM, pp. 87-98, 2023.

[5] Y. Zhang, J. Yin, X. Zhu, and C. Zhang, "Text sentiment classification model based on deep learning," Multimedia Tools and Applications, vol. 78, no. 1, pp. 557–573, 2019, doi: 10.1007/s11042-018-6268-1

[6] A. H. H. K. Walaa Medhat, "Sentiment analysis algorithms and applications," Ain Shams Engineering Journal, vol. 5, no. Ain Shams University, pp. 1093-1113, 2014.

[7] A. Kusumaningtyas and V. Purnamasari, "Indonesian user sentiment analysis for e-wallet application using TF-IDF and SVM," Journal of Information Systems Engineering and Business Intelligence, vol. 7, no. 2, pp. 141–148, 2021.

[8] Gabriella, "PERILAKU MASYARAKAT DI ERA DIGITAL (STUDI DI DESA WATUTUMOU III KECAMATA KALAWAT KABUPATEN MINAHASA UTARA)," Journal Administrasi Publik, vol. 6, no. UNSRAT, pp. 100-112, 2020.

[9] B. H. I. G. P. A. J. Z. Abd. Samad Hasan Basari, "Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization," Procedia Engineering, vol. 53, no. Universiti Malaysia, pp. 453-462, 2013.

[10] C. J. C. A. K. Santra, "Genetic Algorithm and Confusion Matrix for Document Clustering," IJCSI, vol. 9, no. Bharathiar University, Coimbatore, pp. 322-328, 2012.

[11] A. A. R. S. S. P. Rifaldo Sitepu, "he Analysis of Support Vector Machine (SVM) on Monthly Covid-19 Case Classification," IJOICT, vol. 8, no. Telkom University, pp. 40-52, 2022.

[12] R. A. Shahzad Qaiser, "Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents," International Journal of Computer Applications, vol. 181, no. Univesiti Utara Malaysia, 2018.

[13] I. D. a. G. Gediga, "Confusion Matrices and Rough Set Data Analysis," IOP Conf. Series: Journal of Physics: Conf. Series, vol. 2019, no. Institut für Evaluation und Marktanalysen, 2019.

[14] E. S. M. A. U. I. M. T. A. M. I. S. &. A. M. H. Ahmad, "Challenges, comparative analysis and a proposed methodology to predict sentiment from movie reviews using machine learning," ICBDAC, vol. 10, pp. 86-91, 2017.

[15] J. C. M. W. Y. &. P. A. Chou, "Expert Systems with Applications Optimizing parameters of Support Vector Machine using fast messy genetic algorithm for dispute classification," Expert Systems With Applications, vol. 41, 2014.