Sentiment Analysis of iPusnas Application Reviews on Google Play Using Support Vector Machine

[1]Febbi Sena Lestari, [2*]Harliana, [3]Muhamat Maariful Huda, [4]Tito Prabowo
[1,2,3]Ilmu Komputer, Universitas Nahdlatul Ulama Blitar
E-mail: [1]febbisena.l@gmail.com, [2]harliana@unublitar.ac.id, huda.maariful@unubliltar.ac.id, tito.prabowo@unublitar.ac.id*
*Corresponding Author

Abstract— The iPusnas application is a digital library platform that allows users to borrow and read books online, free, and legally. Digital libraries provide literacy facilities without having to come to the library, take part in reducing the spread of the Covid-19 virus, and one of the government's efforts to increase reading interest in Indonesia. User experience in using the application usually conveyed in form of positive or negative reviews on Google Play which can be used to determine the quality of an application. However, too many reviews often make it difficult for readers to conclude the contents of the reviews. Based on that, this research aims to determine user sentiment of iPusnas application by conducting sentiment analysis using Support Vector Machine. The dataset used is iPusnas application reviews on Google Play with 6.946 clean data. Classification produces 94,24% accuracy, 92,38% precision, 83,86% recall, and 87,82% f1-score. This application has 75.1% positive sentiment and 24.9% negative sentiment with a list of frequently appearing words displayed in wordcloud. The final result of this analysis can be used as an evaluation to determine the steps for developing the application to be more optimal so that it can provide a better user experience.

Keywords—sentiment analysis; iPusnas; Support Vector Machine

Corresponding Author:
Harliana,
Ilmu Komputer,
Universitas Nahdlatul Ulama Blitar,
Email: harliana@unublitar.ac.id

I. INTRODUCTION

The iPusnas application is a digital library platform that allows users to borrow and read books like in a library online, free of charge, and legally. This application, made by the National Library of the Republic of Indonesia, can be accessed through websites and mobile-based applications(Maulana, 2018). Nowadays, digital libraries play an important role in providing effective and efficient literacy facilities without having to come to the library and also take part in reducing the spread of the Covid-19 virus(Yois & Marlini, 2020). During the pandemic, access restrictions and closures of public places such as libraries must indeed be circumvented by optimizing the use of technology, one of which is the iPusnas application, to make it easier for people from various circles to add insight.

From credible sources. The presence of this application is also one of the government's efforts to increase reading interest in Indonesia. Users of the iPusnas application, especially those based on Android, are given a simple application interface with various features. User experience in using the application certainly generates various positive and negative comments. These comments are usually submitted in the form of reviews on Google Play, which is one of the most popular platforms for downloading Android apps. These reviews can be used to determine the quality of an application and both its advantages and disadvantages. However, the number of reviews that are too many often makes it difficult for readers to conclude the contents of the reviews, so sentiment analysis is needed to group reviews into positive and negative sentiments (Hanifah, Indriati, & Marji, 2019). In addition, application developers receive evaluations so that they can take the right steps in developing their applications(Sudrajat, Atika, & Herlawati, 2021). This study aims to determine user sentiment towards the iPusnas application. The benefit of this research is that it makes it easier to group iPusnas application reviews into positive and negative sentiments.

Research on application review sentiment analysis has been carried out by several previous researchers using different algorithms. Naive Bayes algorithm has been used for sentiment analysis of video conference application reviews with a mediocore accuracy value of 69% caused by dataset imbalances and devicient preprocessing results(Suhendra, Swastika, & Subianto, 2021). Another study used the K-Nearest Neighbor (K-NN) algorithm on sentiment analysis of the Bibit and Bareksa application reviews with the highest accuracy values of 85.14% and 81.70%, respectively, because they had to do several experiments to determine the appropriate *k* parameter(Adhi Putra, 2021).

In contrast to some of these studies, in this study the researchers will try to increase the accuracy generated in sentiment analysis of the iPusnas application review using the Support Vector Machine (SVM) algorithm. The final result of this analysis can be used as an evaluation to determine the steps for developing the application to be more optimal so that it can provide a better user experience. The Support Vector Machine algorithm was chosen because it learns patterns quickly and gives a high value of accuracy(Budianto, Maryono, & Ariyuana, 2019). This method includes algorithms with good classification performance because it is able to produce high AUC values up to 1,000.(Putri & Kharisudin, 2022). In addition, the execution time required for the Support Vector Machine algorithm is relatively short, in the range of 0.1 seconds to less than 0.3 seconds. (Nur, Setiawan, & Bachtiar, 2019). In conducting sentiment analysis, the Support Vector Machine algorithm has been compared with the Naïve Bayes and K-Nearest Neighbor algorithms, namely online transportation sentiment analysis, with results showing that the Support Vector Machine has the highest accuracy value of 98% and AUC of 0.988 (Rahmatullah, Budiyono, & Saputra, 2021).

II. RESEARCH METHOD

In this research, sentiment analysis was carried out from reviews of the iPusnas application on Google Play using the Support Vector Machine algorithm with the research flow shown in Figure

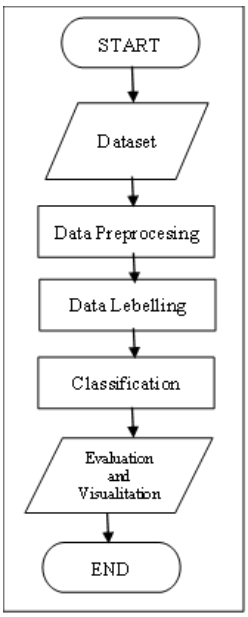

Figure 1. Research Flow

Dataset

The dataset used in this study is the iPusnas application review data found on the Google Play site. The data is taken through scraping technique with Python programming language using Google-Play-Scraper library. The data taken from user reviews of the iPusnas application in the form of comments as of May 24, 2022. The scraping results are stored in a file with .csv extension for simplify analysis.

Preprocessing Data

Basically, the results of data scraping are still unstructured raw data, so they need to be converted into structured data to make them easier to understand. Sentiment analysis using data that has gone through a preprocessing process can provide classification results with better accuracy. The stages of data preprocessing carried out in this study are as follows:

a. *Case Folding*, i.e. change the entire text to lowercase.
b. *Data Cleaning*, i.e. clearing text of unneeded components, such as numbers, punctuation, whitespace, single characters, and non ASCII characters such as emojis.
c. *Tokenizing*, i.e. separating data in the form of sentences into tokens or per word.
d. *Stopword removal*, i.e. eliminating words that are considered meaningless.
e. *Normalization*, i.e. homogenizing words that have the same meaning, but are written differently because of writing errors.
f. *Stemming*, i.e. removing the affixes for each word so that it becomes the basic word.
g. *Join Text*, i.e. returning separate words in token form into whole sentences.
h. *Duplicate Removal*,i.e. namely eliminating duplicate or duplicate data in the document.
i. *Translating*, i.e. translating Indonesian reviews into English.

Data Labelling

Data labeling is the process of labeling review data into positive or negative sentiment classes. Labeling is done automatically using Python's VADER. VADER is only able to label data in English, so the preprocessed data must be translated into English first. VADER produces a compound score that shows the sentiment of the review, namely negative, neutral, and positive. Compound score values of 0 represent negative reviews, compound score values of = 0 represent neutral reviews, and compound score values greater than 0 represent positive reviews.

Classification

Data that has gone through the preprocessing, labeling, and term weighting stages can be classified using the Support Vector Machine algorithm. The dataset to be classified is divided into training data and test data with a data sharing ratio of 80:20, which is 80% training data and 20% test data. The amount of training data that is more than the test data can lead to better data modeling so it gives a good accuracy value. Before classification, the data needs to go through a term weighting process. Textual data needs to be transformed into numeric data so that it can be

classified by a computer. The weighting term used in this study is TF-IDF (Term Frequency-Inverse Document Frequency). The TF-IDF value of a term *t* is the product of the *tf* value and the *idf* value. TF is the frequency of words that appear in a document, while IDF is the number of documents containing each word(Septian, Fahrudin, & Nugroho, 2019). The TF and IDF formulas are shown in equations (1) and (2).

$$tf_{ij} = \frac{f_d(i)}{\max_{j \in d} f_d(j)} \quad (1)$$

$$idf(t, D) = \log\left(\frac{N}{df(i)}\right) \quad (2)$$

The classification concept of the Support Vector Machine algorithm is to find the best hyperplane as a data class separator by maximizing the distance between classes. The kernel function used in this classification is a linear kernel. The formula for minimizing the value on the Support Vector Machine is shown in equation (3) with the conditions listed in equation (4).

$$\frac{1}{2}\|w\|^2 \quad (3)$$

$$y_i(w.x_i + b) \geq 1, \qquad i = 1, 2, 3, \ldots, N \quad (4)$$

Evaluation And Visualization

This research produces two outputs, namely the results of the classification evaluation and visualization. The evaluation results are made in the form of a confusion matrix to determine the values of *accuracy*, *precision*, *recall*, and *f1-score*. The number of true and false test data is classified into four parts, including True Positive (TP) which is positive data that is predicted to be correct as positive data, True Negative (TN) is negative data that is predicted to be correct as negative data, False Positive (FP) which is negative data that is predicted to be wrong as positive data, and False Negative (FN) is positive data that is predicted to be wrong as negative data.(Normawati & Prayogi, 2021).

Visualizations are made in the form of wordcloud and pie charts. Wordcloud displays words that often appear on positive and negative sentiments visually for easy understanding. The larger the font size on the wordcloud, the higher the frequency with which the word appears in the review. Meanwhile, the pie chart shows the percentage of occurrences of each sentiment in the review.

III. RESULT AND DISCUSSION

The dataset in the form of user reviews of the iPusnas application on Google Play was taken using a scraping technique of 11,489 data as of May 24, 2022. The scraping data has 4 attributes, namely at (date of review), username (username), score (review score), and content (user comments) with sample data shown in Table 2. Sentiment analysis only requires user comments so that the data attribute used is content.

Table 2. Result of data *scrapping*

At	UserName	Score	Content
24/05/2022 00:42	Hafidh Mubashir	2	Keluar sendiri tolong diperbaiki
23/05/2022 13:59	Arif Billah	3	Permisi admin sekedar info, aplikasi ini eror. buku yang saya pinjam filenya tidak ditemukan, jadi tidak bisa dibuka ðŸ™•
23/05/2022 06:34	Evalinda Wibowo	4	Saya sudah signup dgn lancar. Kemudian ketika saya akan membaca buku (sy berhasil meminjam) saya hanya stuck….
22/05/2022 17:19	Michael Sihombing	1	Bukunya okelah, tapi APP nya kek sampah kadang gak ada masalah stuck aja di loga….
22/05/2022 14:28	clover clover	1	sdh daftar, sekalinya login gk bisa, yg passwordnya salah lah, username atau email gk valid lah, mau atur ulang kata sandi….

Data Preprocessing

Data preprocessing is done to convert raw data into clean data that is ready to be processed at the next stage. This stage is usually ended by the stemming process, but in this study the stemming data was translated into English so that it could be labeled automatically. The data that have undergone the preprocessing stage are 8,752 data with the sample shown in Table 3.

Table 3. Preprocessing data

Content	Stemmed	Translated
Saya sudah signup dgn lancar. Kemudian ketika saya akan membaca buku (sy berhasil meminjam) saya hanya stuck ditulisan download.... Mohon bantuannya, karena ini sangat mempersulit saya. Terimakasih. *Edit: setelah 1 hari, saya sudh tidak stuck	signup lancar baca buku hasil pinjam stuck tulis download bantu sulit terima kasih edit stuck baca terima kasih depan ipusnas	signup smoothly read borrowed books stuck write download help difficult thank you edit stuck read thank you in front of Ipusnas

lagi sudah bisa membaca... Terimakasih, semoga kedepannya lebih baik lagi iPusnas		
ribet banget cara registrasi, gagal terus	ribet registrasi gagal	complicated registration failed
Aplikasinya bagus dan membantu banget untuk pembaca yang belum sempat membeli dan membaca buku fisiknya. Semoga kedepannya ada fitur wishlist, supaya buku-buku yang 'baru mau'/minat untuk dibaca bisa disimpan dengan baik dan gak takut bakal lupa judul atau susah nyari nanti waktu bener-bener ada waktu buat baca :).	aplikasi bagus bantu baca beli baca buku fisik depan fitur wishlist buku buku minat baca simpan takut lupa judul susah cari baca	good application helps read buys reads physical books in front of the book wishlist feature books interest in reading saves fear of forgetting the title hard to find reading

Data labelling

Data that has been translated into English is automatically labeled using VADER into positive, neutral, and negative sentiment classes. The data that will be processed in the next stage is data that is labeled positive and negative so that data labeled neutral is removed. The net data that can be classified are 6,946 data with the sample shown in Table 4.

Table 4. labelisasi data

Review	Compound_score	Label
signup smoothly read borrowed books stuck write download help difficult thank you edit stuck read thank you in front of Ipusnas	0.296	Positive
complicated registration failed	-0.5106	Negative
good application helps read buys reads physical books in front of the book wishlist feature books interest in reading saves fear of forgetting the title hard to find reading	0.5994	Positive

Classification

Sentiment classification is done using the Support Vector Machine algorithm with a linear kernel function. Classification is done with the Python programming language using Google

Collaboratory. Before being classified, the dataset contains 6,946 data divided into training data and test data with a division ratio of 80:20 so that there are 5,556 training data and 1,390 test data. The labeled training data goes through a term weighting stage with TF-IDF to convert it into numeric data. The weighting of terms in the training data forms a dataframe with 5,556 rows × 4,001 columns which indicates that there are 4,001 unique terms in the 5,556 data training. The term weighted sample is shown in Table 5.

Table 5. Sample of term weighting results

Aaaw	Abarat	Abc	Abdullah	…	Zoom	Zoomed
0.0	0.0	0.0	0.0	…	0.0	0.0
0.0	0.0	0.0	0.0	…	0.0	0.0
0.0	0.0	0.0	0.0	…	0.0	0.0
…	…	…	…	…	…	…
0.0	0.0	0.0	0.0	…	0.0	0.0
0.0	0.0	0.0	0.0	…	0.0	0.0
0.0	0.0	0.0	0.0	…	0.0	0.0

Classification with the Support Vector Machine algorithm is done by calling the LinearSVC function in Python. However, the value of the C parameter to be used needs to be determined first. Determination of the value of C shown in Figure 2 can be done by trying several values on the Support Vector Machine algorithm model. Among the rows of values entered, the value of 0.75 produces the highest accuracy of 0.9424 or 94.24% so that it can be used as the value of the C parameter in the algorithm.

The Support Vector Machine algorithm model with a value of C = 0.75 trained 5.556 training data in the form of attributes and labels which resulted in a final model accuracy value of 0.9424 or 94.24% as shown in Figure 3. The process can be continued with model evaluation that displays more accuracy values detail

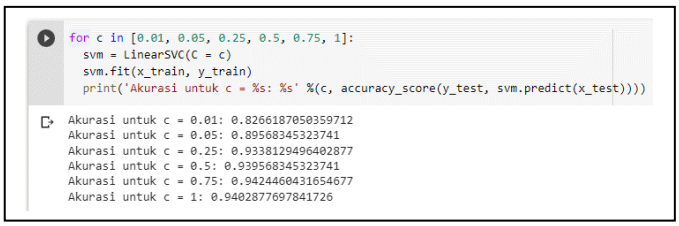

Figure 2. Determining the value of the c parameter in Python

```
svm = LinearSVC(C = 0.75)
svm.fit(x_train, y_train)

print('Accuracy score model final: %s' %accuracy_score(y_test, svm.predict(x_test)))

Accuracy score model final: 0.9424460431654677
```

Figure 3. *Accuracy score model final* in Python

Evaluation And Visualitation

Datasets that have been classified using the Support Vector Machine algorithm are evaluated by the confusion matrix shown in Table 6. The values contained in the confusion matrix can be used to calculate accuracy, precision, recall, and f1-score values. Based on these calculations, the classification with the Support Vector Machine produces an accuracy value of 94.24%, precision 92.38%, recall 83.86%, and f1-score 87.82% which means that it is quite good in sentiment analysis of the iPusnas application review.

Table 6. *Confusion matrix* table

Class	***Positive***	***Negative***
Positive	291	56
Negative	24	1.019

Pie chart is made to find out the percentage of occurrences of each sentiment from a dataset that has been labeled. The sentiment class pie chart in Figure 2 shows that the iPusnas application has 75.1% or 5,215 positive sentiment data and 24.9% or 1,731 negative sentiment data. The words that often appear in each sentiment are displayed in the form of a word cloud. The word cloud of positive sentiment in Figure 3 shows the words with the highest frequency of occurrence in the positive sentiment class are “book”, “application”, “good”, “ipusnas”, and “read”. Meanwhile, the words with the highest frequency of occurrence in the negative sentiment class shown in Figure 4 are “book”, “application”, “error”, “login”, and “open”.

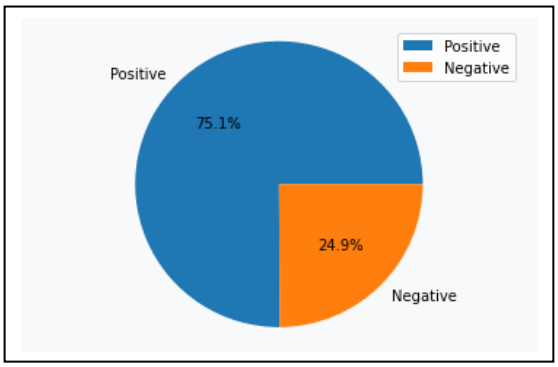

Figure 3. *Pie chart* Sentiment Class

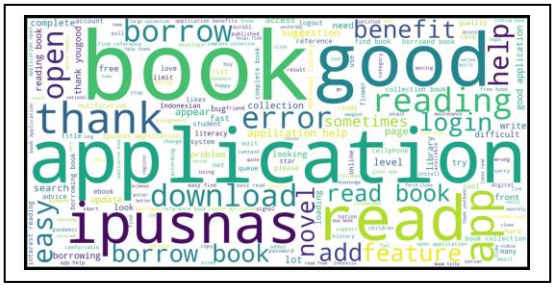

Figure 4. *Wordcloud* form positive sentiment

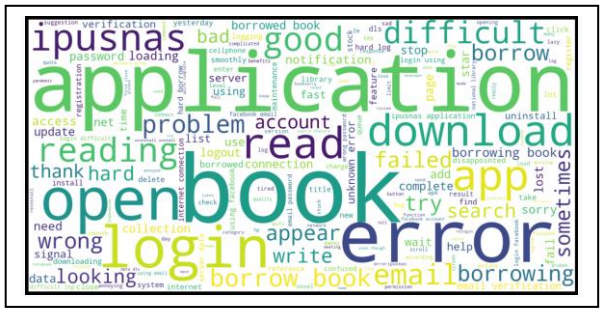

Figure 5. *Wordcloud* from negative sentiment

IV. CONCLUSION

According to the research that has been done, the results show that the sentiment analysis of the iPusnas application review with the Support Vector Machine provides a relatively good accuracy value. Classification performed on 6,946 clean data resulted in 94.24% accuracy, 92.38% precision, 83.86% recall, and 87.82% f1-score. This application has 75.1% positive sentiment and 24.9% negative sentiment with a list of frequently appearing words displayed in wordcloud form. The final result of this analysis can be used as an evaluation to determine the steps for developing the application to be more optimal so that it can provide a better user experience.

REFERENCES

Adhi Putra, A. D. (2021). Analisis Sentimen pada Ulasan pengguna Aplikasi Bibit Dan Bareksa dengan Algoritma KNN. JATISI (Jurnal Teknik Informatika Dan Sistem Informasi), 8(2), 636–646. https://doi.org/10.35957/jatisi.v8i2.962

Budianto, A., Maryono, D., & Ariyuana, R. (2019). Perbandingan K-Nearest Neighbor (KNN) dan Support Vector Machine (SVM) Dalam Pengenalan Karakter Plat Kendaraan Bermotor. Jurnal Ilmiah Pendidikan Teknik Dan Kejuruan, 11(1), 27. https://doi.org/10.20961/jiptek.v11i1.18018

Hanifah, S., Indriati, & Marji. (2019). Analisis Sentimen Kepuasan Pengguna Pada Ulasan

Aplikasi Marketplace Menggunakan Metode BM25F dan Neighbor-Weighted K-Nearest Neighbor. Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer, 3(10), 10309–10315.

Maulana, Y. I. (2018). Evaluasi Tingkat Kepuasan Pengguna Perpustakaan Digital Nasional (iPusnas) dengan Kerangka PIECES. Bianglala Informatika, 6(1), 51–55. https://doi.org/https://doi.org/10.31294/bi.v6i1.5904

Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. Jurnal Sains Komputer & Informatika (J-SAKTI), 5(2), 697–711.

Nur, I. T. A., Setiawan, N. Y., & Bachtiar, F. A. (2019). Perbandingan Performa Metode Klasifikasi SVM, Neural Network, dan Naive Bayes untuk Mendeteksi Kualitas Pengajuan Kredit di Koperasi Simpan Pinjam. Jurnal Teknologi Informasi Dan Ilmu Komputer, 6(4), 445–450. https://doi.org/10.25126/jtiik.2019641352

Putri, M. I., & Kharisudin, I. (2022). Analisis Sentimen Pengguna Aplikasi Marketplace Tokopedia Pada Situs Google Play Menggunakan Metode Support Vector Machine (SVM), Naïve Bayes , dan Logistic Regression. 5, 759–766.

Rahmatullah, B., Budiyono, P., & Saputra, S. A. (2021). Sentimen Analisis Transportasi Online Menggunakan Algoritma Support Vector Machine, Naive Bayes, dan KNN. Jurnal Ilmu Komputer JIK, IV(02), 9–15.

Septian, J. A., Fahrudin, T. M., & Nugroho, A. (2019). Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor. Journal of Intelligent Systems and Computation, 1(1), 43–49.

Sudrajat, M. R., Atika, P. D., & Herlawati. (2021). Implementasi Support Vector Machine (SVM) dan Naive Bayes untuk Analisis Sentimen Aplikasi KAI Access. Jurnal ICT : Information Communication & Technology, 20(2), 254–259.

Suhendra, M., Swastika, W., & Subianto, M. (2021). Analisis Sentimen Pada Ulasan Aplikasi Video Conference Menggunakan Naive Bayes. Sainsbertek Jurnal Ilmiah Sains & Teknologi, 2(1), 1–9. https://doi.org/10.33479/sb.v2i1.145

Yois, N. P., & Marlini. (2020). Sumber Daya Akses Perpustakaan Digital pada Masa Pandemi Corona. BIBLIOTIKA : Jurnal Kajian Perpustakaan Dan Informasi, 4(2), 218–222.