Indonesian Journal of Digital Business

Journal homepage: https://ejournal.upi.edu/index.php/IJDB/index

Sentiment Analysis For User Review Classification On Jenius Application Using Naive Bayes Algorithm Based On Particle Swarm Optimization

Vivi Indah Fitriani[1], Oding Herdiana[2], Rangga Gelar Guntara [3]

Universitas Pendidikan Indonesia, Indonesia

Correspondence: E-mail: viviindahfitriani@gmail.com

ABSTRACT

The rapid development of information technology and communication has facilitated various aspects of life, including the banking sector. Jenius is one of the digital banking applications that has been downloaded by five million users and continues to grow. With the increasing number of users, there is a growing number of opinions written about their experiences using the application, making it difficult to identify reviews through manual monitoring without textual data processing. This study aims to classify user reviews of the Jenius application on Google Playstore using the Naive Bayes algorithm and Particle Swarm Optimization feature selection. The data used consists of 3047 user reviews of the Jenius application collected from January 16, 2022 to April 13, 2023 and will be divided into two classes, namely positive and negative sentiment. This study also compares the Naive Bayes algorithm using PSO feature selection and without using PSO feature selection. The test results of the two methods indicate that the PSO feature selection with 800 iterations proves to be effective in optimizing the performance of the Naive Bayes algorithm model with an accuracy of 98.50%, precision of 97.81%, recall of 99.36%, and F1-score of 98.58%. Meanwhile, the performance level of the Naive Bayes algorithm without using PSO feature selection is lower with an accuracy of 96.68%, precision of 94.83%, recall of 99.04%, and F1-score of 96.88%.

Informasi Artikel

Article Histsory:
Accepted 20 April 2024
Revised 25 April 2024
Available Online 24 May 2024
Publication 24 May 2024

Keyword*:*
Sentiment Analysis, Naive Bayes, Particle Swarm Optimization, Jenius

1. INTRODUCTION

In the era of the current industrial revolution 4.0, where the development of information and communication technology is rapidly advancing, it facilitates various aspects of life. One field that follows the wave of digitalization is the banking industry, which has now become digital banking. It is undeniable that digital banking plays a crucial role in improving the economy of a country by providing convenience to its customers (Susilawaty & Nicola, 2020). With digital banking, customers can obtain information and perform transactions beyond traditional banking products, such as investment information, e-commerce transactions, financial advisory services, and many other services that can be accessed through smartphones (Moestopo et al., 2019).

Jenius is a digital banking application initiated by PT BTPN Tbk. The application was released on August 11, 2016, and was the first digital bank pioneer in Indonesia, offering an easy online registration and account opening process through a smartphone application. The Jenius application has been downloaded by five million users from the Google Play Store and has received 188,000 user reviews. According to idx.channel, PT BTPN Tbk reported that the number of active Jenius application users increased to 3.99 million customers, representing a 19% growth in June 2022 compared to 3.34 million customers in June 2021 (Rabbi, 2022). With the increasing number of users, there is a growing amount of opinions written by users, making it challenging to identify the strengths and weaknesses of the application through manual monitoring without textual data processing.

Sentiment analysis is a branch of text mining that focuses on classifying texts or documents based on public opinions or comments about a particular topic, determining whether the comments have positive or negative sentiment (Sari & Wibowo, 2019). Sentiment analysis can help companies obtain feedback on their brands and assist the general public in evaluating products based on the latest thoughts and reviews (Gunawan et al., 2018).

One algorithm that has proven to be an efficient and effective method for sentiment analysis classification is the Naive Bayes algorithm (Hendra, 2021). Naive Bayes algorithm has advantages compared to other classification algorithms, as it requires only a small amount of training data to estimate parameters in the classification process (Maruli Sitompul et al., 2021). It has a simple calculation yet has been proven to achieve high accuracy and fast learning process (Prajamukti & Mega Santoni, 2021). However, Naive Bayes algorithm also has some weaknesses, such as the strong assumption of feature independence in the text, which can affect the accuracy level. (Putri et al., 2020). To overcome this, the PSO technique is needed, which can be used to enhance the performance of the Naive Bayes algorithm by optimizing parameters or weights in the classification process.

PSO has been proven to be an effective optimization technique for addressing complex optimization problems involving multiple dimensions and parameters in various machine learning applications, including neural networks, SVM, Naive Bayes and other classification algorithms (Herdiana et al., 2021). However, the use of PSO in the context of sentiment analysis on user reviews of mobile banking applications such as Jenius is still limited. Therefore, the objective of this research is to conduct sentiment analysis on user reviews of the Jenius application using the Naive Bayes algorithm based on Particle Swarm Optimization. The combination of these two algorithms is expected to enhance the accuracy in classifying user reviews into appropriate sentiment categories, and the results of this study can contribute to the knowledge for the better development of the Jenius application.

2. METHODOLOGY

This research uses the Cross-Industry Standard Process for Data Mining (CRISP-DM) method. CRISP-DM is a standard data mining process that is commonly used as a problem-solving strategy in business and research. There are 6 stages in the CRISP-DM method, namely business understanding, data understanding, data preparation, modelling, evaluation, and deployment (Daniel T.Larose, 2015).

a) Business Understanding

In this phase, problem identification is carried out based on the predetermined topic, which is the classification of user reviews on the Jenius application using the Naive Bayes algorithm. This process aims to determine the objectives and limitations of the problem within the scope of this research.

b) Data Understanding

This phase aims to collect, identify, and understand the existing data. It is essential that the data can be verified for its accuracy. The data utilized in this study comprises user reviews obtained from the Play Store, specifically related to two online investment applications known as Bibit and Bareksa.

c) Data Preparation

In this phase, the data preparation process is carried out with the aim of obtaining clean data and ready to use for the research. This stage is often used to transform previously unstructured data into structured data. The following stages in data preparation are data cleaning, case folding, normalization, stopwords removal, tokenization, and stemming.

d) Modelling

In this phase, the process of building a classification model using the Naive Bayes algorithm and Particle Swarm Optimization (PSO) feature selection is performed. However, before classification, the data will be divided into training data and testing data, with an 80% proportion for training data and a 20% proportion for testing data. Based on research (Yunhasnawa & Mardhika, 2019), a system will perform better in identifying patterns in new data if the proportion of training data is larger than the proportion of testing data.

e) Evaluation

In this phase, evaluation is conducted using confusion matrix to measure the performance of the Naive Bayes algorithm and PSO feature selection model that has been formed. The classification performance will be measured by calculating the accuracy, precision, recall, and F1-score.

f) Deployment

After the modeling process, the next phase is the deployment stage, where the final report is prepared based on the evaluated classification results. The final report includes data visualization in the form of word clouds and graphs to facilitate readers in understanding the research findings. Wordcloud visualization aims to identify the most frequently appearing words in each positive and negative class.

3. RESULT AND DISCUSSION

3.1. Business Understanding

The purpose of this business understanding is to analyze the sentiment determine the outcomes of positive and negative sentiments that will be classified using the Naive Bayes algorithm and optimized using the Particle Swarm Optimization technique. Additionally, it aims to identify the requirements and limitations of performing sentiment analysis on user reviews of the Jenius application.

3.2 Data Understanding

In this phase, an understanding of the data used was conducted. The researcher collected user review data of the Jenius application from January 16, 2022, to April 13, 2023. The obtained data included usernames, dates, scores (ratings), and content. The data scraping process resulted in 3047 records of review data, with 1710 positive reviews and 1337 negative reviews. The explanation of the variables used in this study is shown in Table 1 below:

Tabel 1. Variable

Variable	**Description**
Username	Username that is used as an account identity and is unique
Score	User's rating of the Jenius app which consists of a score of 1-5
At	Date and time of user review submission.
Content	Reviews given by users about the Jenius apps

3.3 Data Preparation

Data preparation in this research begins with data preprocessing, which consists of six processes: data cleaning, case folding, tokenizing, normalization, stopwords, and stemming. After the data is cleaned, data labeling and TF-IDF word weighting are performed.

a. Data Cleaning

This process is conducted to remove noise from the data, such as punctuation marks, symbols, numbers, and emoticons, using the RegEx (Regular Expression) library. The results of the data cleaning stage can be seen in Table 2 below.

Tabel 2. Data Cleaning Result

Before *Cleaning*	**After *Cleaning***
Pelayanan CS jelek, aplikasinya ngebug gabisa daftar.. Hati2 aja kalau mau menyimpan duit disini takutnya ada bug duitnya ilang	Pelayanan CS jelek aplikasinya ngebug gabisa daftar Hati aja kalau mau menyimpan duit disini takutnya ada bug duitnya ilang

b. Case Folding

Case Folding is the process of standardizing characters by converting all uppercase letters in the document to lowercase using the "lower" library in Python. The results of the case folding stage can be seen in Table 3 below.

Tabel 3. Case Folding Result

Before *Case Folding*	**After *Case Folding***
Pelayanan CS jelek aplikasinya ngebug gabisa daftar Hati aja kalau mau menyimpan duit disini takutnya ada bug duitnya ilang	pelayanan cs jelek aplikasinya ngebug gabisa daftar hati aja kalau mau menyimpan duit disini takutnya ada bug duitnya ilang

c. Tokenizing

This process involves splitting sentences in the review text into several part of word using whitespace as the separator. The results of the tokenizing stage can be seen in Table 4 below.

Tabel 4. Tokenizing Result

Before *Tokenizing*	**After *Tokenizing***
pelayanan cs jelek aplikasinya ngebug gabisa daftar hati aja kalau mau menyimpan duit disini takutnya ada bug duitnya ilang	['pelayanan', 'cs', 'jelek', 'aplikasinya', 'ngebug', 'gabisa', 'daftar', 'hati', 'aja', 'kalau', 'mau', 'menyimpan', 'duit', 'disini', 'takutnya', 'ada', 'bug', 'duitnya', 'ilang']

d. Normalization

This process involves converting non-standard, abbreviations, or misspelled words into standard words according to the Indonesian dictionary (KBBI). The results of the normalization stage can be seen in Table 5 below.

Tabel 5. Normalization Result

Before *Normalization*	**After *Normalization***
['pelayanan', 'cs', 'jelek', 'aplikasinya', 'ngebug', 'gabisa', 'daftar', 'hati', 'aja', 'kalau', 'mau', 'menyimpan', 'duit', 'disini', 'takutnya', 'ada', 'bug', 'uangnya', 'ilang']	['pelayanan', **'customer service'**, 'jelek', 'aplikasinya', 'ngebug', **'tidak bisa'**, 'daftar', 'hati', **'saja'**, 'kalau', **'ingin'**, 'menyimpan', **'uang'**, 'disini', 'takutnya', 'ada', 'bug', '**uangnya**', **'hilang'**]

e. Stopwords

This process involves removing insignificant words for classification using the Sastrawi stopwords list. The results of the stopwords stage can be seen in Table 6 below

Tabel 6. Stopwords Result

Before *Stopwords*	**After *Stopwords***
['pelayanan', 'customer service', 'jelek', 'aplikasinya', 'ngebug', 'tidak bisa', 'daftar', 'hati', 'saja', 'kalau', 'ingin', 'menyimpan', 'uang', 'disini', 'takutnya', 'ada', 'bug', 'uangnya', 'hilang']	['pelayanan', 'customer service', 'jelek', 'aplikasinya', 'bug', 'tidak bisa', 'daftar', 'hati', 'menyimpan', 'uang', 'takutnya', 'bug', 'uangnya', 'hilang']

f. Stemming

After the stopwords removal stage, the stemming process is applied. This process involves converting words with affixes into their base form.

Tabel 7. Stemming Result

Before *Stemming*	**After *Stemming***
['pelayanan', 'customer service', 'jelek', 'aplikasinya', 'bug', 'tidak bisa', 'daftar', 'hati', 'menyimpan', 'uang', 'takutnya', 'bug', 'uangnya', 'hilang']	[**'layan'**, 'customer service', 'jelek', **'aplikasi'**, 'bug', 'tidak bisa', 'daftar', 'hati', 'simpan', 'uang', **'takut'**, 'bug', 'uang', 'hilang']

g. Data Labelling

After the pre-processing stage, the obtained data are then manually labelled into two classes: positive and negative sentiment. Out of a total of 3011 cleaned data, the manual labeling resulted in 1710 positive review data and 1301 negative review data. The user sentiment towards the Jenius application is dominated by positive sentiment at 56.8%, while negative sentiment is 43.2%.

Tabel 8. Data Labelling

Ulasan	**Label**	***Label Encoder***
Parah cuma gara² ktp saya kusam ga bisa sama sekali buat aktivasi, vidio call sampe 6 kali tetap saja ga bisa	Negatif	0
aplikasi jenius terbaik, UI UX simple, memudahkan untuk topup ewallet, kartu debit yang memudahkan untuk berlangganan netflix dan spotify	Positif	1

h. Term Frequency – Inverse Document Frequency

After performing data cleaning in the text preprocessing stage, the next step is to perform word weighting on each sentence using the TF-IDF Vectorizer from the scikit-learn library. TF-IDF assigns weights or values to words based on their frequency of occurrence in each document, which will be used as features in the classification process. Below are the results of the TF-IDF calculations:

Tabel 9. TF-IDF

Term	TF-IDF				
	D1	D2	D3	D4	D5
susah	0.699	0	0	0	0
login	0.222	0.222	0.222	0	0
aplikasi	0.097	0.097	0	0.097	0.097
error	0	0.699	0	0	0
sulit	0	0	0.699	0	0
nomor	0	0	0.699	0	0
ponsel	0	0	0.699	0	0
email	0	0	0.699	0	0
daftar	0	0	0.699	0	0
masuk	0	0	0.699	0	0
pin	0	0	0.699	0	0
tidak bisa	0	0	0.699	0	0
jenius	0	0	0	0.398	0.398
baik	0	0	0	0.699	0
user interface	0	0	0	0.699	0
user experience	0	0	0	0.699	0
simpel	0	0	0	0.699	0
mudah	0	0	0	0.444	0.222
topup	0	0	0	0.398	0.398
electronic wallet	0	0	0	0.398	0.398
kartu	0	0	0	0.699	0
debit	0	0	0	0.699	0
langgan	0	0	0	0.699	0
netflix	0	0	0	0.699	0
spotify	0	0	0	0.699	0
fitur	0	0	0	0	0.699
lengkap	0	0	0	0	0.699
tabung	0	0	0	0	0.699
nyaman	0	0	0	0	0.699

3.4 Modelling

a. Splitting Data

After performing TF-IDF word weighting, the next step is to divide the data into two sets: the training set and the testing set with a ratio of 80% for training data and 20% for testing data, based on the total of 3011 preprocessed data. The detailed distribution of the training and testing data can be seen in Table 11.

Tabel 11. Splitting Data

Jenis Data	Presentase	Jumlah
Training	80%	2408
Testing	20%	603
Jumlah	**100%**	**3011**

b. Naive Bayes Classification

After dividing the data into training and testing sets, the next step is to build model for the classification process. In this study, the algorithm used for training is Multinomial Naive Bayes. The training process in this study uses the data samples from the document's TF-IDF word weighting results in Table 4.10 mentioned earlier. After obtaining the TF-IDF results as shown in Table 4.1, the next step is to calculate the probabilities of each class (prior) in the training data using the following equation:

$$P(C) = \frac{|docs\ c|}{|documents|}$$

$$P(Positif) = \frac{2}{5} = 0,4$$

$$P(Negatif) = \frac{3}{5} = 0,6$$

The next step is to calculate the likelihood values for positive and negative classes, as shown in Table 11, using Equation (2) as explained in subsection (1.8).

Tabel 12. Likelihood Probability

Term	Probabilitas	
	P(X|Negatif)	*P(X|Positif)*
susah	0.0386	0.0192
login	0.0379	0.0192
aplikasi	0.0271	0.023

error	0.0386	0.0192
sulit	0.0386	0.0192
nomor	0.0386	0.0192
ponsel	0.0386	0.0192
email	0.0386	0.0192
daftar	0.0386	0.0192
masuk	0.0386	0.0192
pin	0.0386	0.0192
tidak bisa	0.0386	0.0192
jenius	0.0227	0.0345
baik	0.0227	0.0327
user interface	0.0227	0.0327
user experience	0.0227	0.0327
simpel	0.0227	0.0327
mudah	0.0227	0.032
topup	0.0227	0.0345
electronic wallet	0.0227	0.0345
kartu	0.0227	0.0327
debit	0.0227	0.0327
langgan	0.0227	0.0327
netflix	0.0227	0.0327
spotify	0.0227	0.0327
fitur	0.0227	0.0327
lengkap	0.0227	0.0327
tabung	0.0227	0.0327
nyaman	0.0227	0.0327

After obtaining the probability values for each word from the training data, the next step is to perform testing on the test data using the Naive Bayes model built on the training data. The following is an example of one out of 603 test data, as shown in Table 13:

Tabel 13. Data Testing Sample

	Data Testing
D1	['susah', 'login', 'aplikasi']

For the test data shown in the table above, the probability of each term occurring will be calculated for each class, resulting in the calculations shown in Table 14:

Tabel 14. Data Testing Probability

Term	**P(X|Positif)**	**P(X|Negatif)**
susah	0.0386	0.0192
login	0.0379	0.0192
aplikasi	0.0271	0.023

After calculating the probabilities for each term, the next step is to determine the class of the test data sample, whether it belongs to the positive or negative class, using Equation in Chapter 2:

P(Wi) = P(Cj) * (P(Wi|Cj)) * ... * P(Wn|Cn)

1. ***P(positif|data uji)***

   *P(positif|data uji) = P(positif) * P('susah'|positif) * P('login'|positif) * P('aplikasi'|positif)*
   = 0.4 * 0.0192 * 0.0192 * 0.023
   = 0.000003391488
2. ***P(Negatif|data uji)***

   *P(negatif | data uji) = P(negatif) * P('susah'|negatif) * P('login' |negatif) * P('aplikasi' |negatif)*
   = 0.6 * 0.0386 * 0.0379 * 0.0271
   = 0.0000237874044

The classification of the test data is determined based on the highest probability value. Based on the calculations above, it can be seen that P(negative|test data) > P(positive|test data) with the highest probability value of 0.0000237874044. Therefore, the test data sample is classified as negative.

c. Particle Swarm Optimization (PSO)

In the feature selection process, several parameter initializations are required. These include learning rates (c1 and c2) with c1 value of 1 and c2 value of 2, inertia weight (w) with a value of 0.9, number of neighbors (k) with a value of 30, Minkowski p-norm (p) with a value of 2, and number of particles (n_particles) with a value of 30. Furthermore, using the same parameter values, different scenarios will be initialized with varying numbers of Particle Swarm Optimization iterations: 50, 100, 250, 500, and 800. The experiments will stop when the maximum number of iterations is reached.

Tabel 15. PSO

Iterasi	Fitur	Best Cost	Akurasi
50	893	0.0216	97.84%
100	876	0.0216	97.84%
250	883	0.0199	98.00%
500	882	0.0166	98.34%
800	**907**	**0.0149**	**98.50%**

Based on the experimental results conducted in five PSO scenarios with different numbers of iterations, as shown in Table 4.17, the scenario with 800 iterations in the PSO feature selection process has the largest number of features, which is 907 features, and achieves the highest accuracy of 98.50%. This accuracy value is higher compared to the classification without using the Particle Swarm Optimization optimization method.

3.5 Evaluation

a. Evaluation of naive bayes algorithm

After the machine learning model using the Naive Bayes algorithm is formed, the next step is to evaluate the model to measure its classification performance using a confusion matrix.

Tabel 16. Confusion Matrix NB

	True Positive	**True Negative**
Pred.Positive	312	17
Pred.Negative	3	271

The calculation from Table 5 is as follows:

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN}$ =$\frac{583}{603}$ = 0,9668

Precision = $\frac{TP}{TP+FP}$ = $\frac{312}{312+17}$ = $\frac{312}{329}$ = 0,9483

Recall = $\frac{TP}{TP+FN}$ = $\frac{312}{312+3}$= $\frac{312}{315}$ = 0,9904

F1 Score = $2x\,\frac{recall*precision}{recall+precision}$ = 0.9688

b. Evaluation of Naive Bayes based on PSO

In the second testing, an evaluation was conducted on the Naive Bayes model using particle swarm optimization with 5 scenarios, and a comparison was made between the two methods, as shown in the table below: From the evaluation calculations, it is known that NB.

Tabel 17. Comparison of Evaluation NB + PSO

Model	Pengukuran Performa			
	Accuracy	Precision	Recall	F1 Score
Naive Bayes	96,68%	94,83%	99,04%	96,88%
NB+ PSO (50 iterasi)	97,84%	96,89%	99,04%	97,95%
NB+ PSO (100 iterasi)	97,84%	96,60%	99,36%	97,96%
NB+ PSO (250 iterasi)	98,00%	96,90%	99,36%	98,11%
NB+ PSO (500 iterasi)	98.34%	97.80%	99.04%	98.42%
NB+ PSO (800 iterasi)	**98.50%**	**97.81%**	**99.36%**	**98.58%**

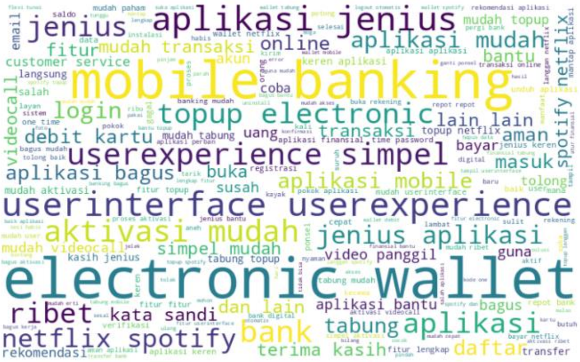
Based on the performance comparison results in Table 4.18 above, it can be seen that the best model performance for sentiment analysis classification using Jenius application review data is the Naive Bayes classification model with Particle Swarm Optimization (PSO) feature selection of 800 iterations which produces better accuracy, precision, recall, and f1-score values compared to Naive Bayes classification without PSO feature selection. From the details of the comparison, it can be concluded that the application of PSO feature selection in this study is able to optimize the performance of the Naive Bayes algorithm classification model. The improvement can be seen in the feature selection experiment in the fifth PSO scenario with a total of 800 iterations with an increase in accuracy value of 1.82%, precision value of 2.98%, recall value of 0.32%, and f-1 score value with an increase of 1.70%.

3.6 Deployment

During the deployment phase, a visualization will be created based on the processed data from previous stages. The visualization takes the form of a word cloud to represent every word that appears in both positive and negative reviews.

a. Wordcloud Positive Sentiment

In Figure 4.8 below is a visualization of the word cloud depicting the positive sentiment of user reviews for the Jenius application, based on the frequency of words that commonly appear in Jenius review data. It can be illustrated as follows:



Gambar 1. Wordcloud Positive Sentiment

Based on Figure 1, it can be observed that the most frequently appearing words in the positive sentiment of user reviews for the Jenius application are "aplikasi", "mudah", "jenius", "mobile banking," "electronic wallet," "user experience," "simpel", "user interface," "top up," "aktivasi", "netflix," "bantu", and others. From the word cloud, it can be inferred that Jenius is a user-friendly mobile banking application that assists users in their financial activities such as saving, investing, credit card payments, and more and It has a simple user interface and user experience, along with attractive features that enable various transactions such as e-wallet top-ups and subscriptions to platforms like Netflix, Spotify, and others.

b. Wordcloud Negative Sentiment

The following is a visualization in the form of a word cloud displaying the frequency of words that commonly appear in the negative sentiment of user reviews for the Jenius application. It can be illustrated in Figure 4.10 as follows:

Gambar 2. Wordcloud Negative Sentiment

Based on Figure 2, it can be seen that the most frequently appearing words in the negative sentiment of user reviews for the Jenius application are "aplikasi", "login," "jenius," "susah", "masuk", "daftar", "kata sandi", "tolong", "akun", "customer service," and "ribet", etc. From these words, it can be concluded that users of the Jenius application has several challenges related to the app, including difficulties in login, the inconvenience of contacting customer service, issues with password resets, and the app being heavy, causing session timeouts and force close occurrences.

4. CONCLUSION & RECOMMENDATIONS

a. Conclusion

Based on the research results, it can be concluded that the implementation of Particle Swarm Optimization feature selection is proven to optimize the performance of the Naive Bayes algorithm model in analyzing the sentiment of Jenius application users. The test results show that PSO feature selection with 800 iterations has succeeded in increasing the accuracy value by 98.50%, precision by 97.81%, recall by 99.36%, and f1-score by 98.58%, compared to the performance level of a single Naive Bayes classification evaluation model, which had an accuracy value of 96.68%, precision by 94.83%, recall by 99.04%, and f1-score by 96.88%. The Utilization of PSO feature selection can discover a better combination of weights or features in the Naive Bayes algorithm so as to classify user reviews more accurately and increase the accuracy value by 1.8%.

b. Recommendations

Further research is expected to increase the dataset size to improve the performance of the classification algorithm model. And data balancing techniques such as SMOTE, ROS and ADSYN can be applied to improve the performance of the classification algorithm.

5. REFERENCE

Gunawan, B., Sasty, H. P., & Esyudha, E. (2018). Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, *4*(2), 17–29. www.femaledaily.com

Hendra, A. (2021). Analisis Sentimen Review Halodoc Menggunakan Naive Bayes Classifier. In *JISKa* (Vol. 6, Issue 2). MEI.

Herdiana, O., Maulani, S., Firdaus, E. A., & Kunci, K. (2021). Strategi Pemasaran Produk Industri Kreatif Menggunakan Algoritma K-Means Clustering Berbasis Particle Swarm Optimization. *Jurnal Nuansa Informatika*, *15*, 1–13. https://journal.uniku.ac.id/index.php/ilkom

Maruli Sitompul, A., Tunas Bangsa, S., Utara, S., Tunas Bangsa, A., & Jln Sudirman Blok No, I. A. (2021). Teknik Data Mining Dalam Prediksi Jumlah Siswa Baru Dengan Algoritma Naive Bayes. *KESATRIA : Jurnal Penerapan Sistem Informasi (Komputer Dan Managemen)*, *2*(2), 108–117.

Moestopo, U., Ayufi, N., Pertiwi, R., & Shabrina, V. G. (2019). Pemanfaatan Co-Creation oleh Jenis Co.Create sebagai Strategi Pengembangan Produk Perbankan Digital Jenius. *Petanda : Jurnal Ilmu Komunikasi Dan Humaniora* , *01*(02), 52–62.

Prajamukti, R., & Mega Santoni, M. (2021, October 28). Klasifikasi Dan Analisis Sentimen Pada Data Twitter Menggunakan Algoritma Naïve Bayes (Studi Kasus: Timnas Indonesia Senior, U-23, Dan U-19). *Seminar Nasional Informatika, Sistem Informasi, Dan Keamanan Siber (SEINASI-KESI)*. https://t.co/SAbaU6Prrz

Putri, D. A., Kristiyanti, D. A., Indrayuni, E., Nurhadi, A., & Hadinata, D. R. (2020). Comparison of Naive Bayes Algorithm and Support Vector Machine using PSO Feature Selection for Sentiment Analysis on E-Wallet Review. *Journal of Physics: Conference Series*, *1641*(1). https://doi.org/10.1088/1742-6596/1641/1/012085

Sari, F. V., & Wibowo, A. (2019). Analisis Sentimen Pelanggan Toko Online Jd.Id Menggunakan Metode Naïve Bayes Classifier Berbasis Konversi Ikon Emosi. *Jurnal SIMETRIS*, *10*(2).

Susilawaty, L., & Nicola, N. (2020). Pengaruh layanan perbankan digital pada kepuasan nasabah perbankan. *Jurnal Manajemen Maranatha*, *19*(2), 179–190. https://doi.org/10.28932/jmm.v19i2.2478

Yunhasnawa, Y., & Mardhika, D. (2019). *Implementasi Metode Backpropagation pada Prediksi Pemakaian Air Perbulan*

*(Studi Kasus: PDAM Kabupaten Malang
Unit Pakisaji)*.

p- ISSN 2798-0014 e- ISSN 2798-2432