Indonesian Journal of Computer Science
ISSN 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam No. 1, Padang, Indonesia
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Comparison of Machine Learning Algorithm Models in Bitcoin Price Sentiment Analysis

Rizky Afrinanda[1], Wahyu Tawa Bagus[2], Lusiana Efrizoni[3]
rizkyafrinandaa@gmail.com[1*], wahyutawabagus@gmail.com[2], lusiana@stmik-amik-riau.ac.id[3]
Department of Computer Science, STMIK AMIK Riau

Article Information

Submitted : 30 Mar 2023
Reviewed: 14 Apr 2023
Accepted : 27 Apr 2023

Keywords

Bitcoin, Feature Selection, Naïve Bayes, Sentiment Analysis, Support Vector Machine

Abstract

Bitcoin is one of the digital payments that is currently booming, fast delivery makes bitcoin in great demand by many people, currently there are many digital currency exchanges that can be used, one of the well-known ones in Indonesia, namely Indodax. Indodax is a cryptocurrency exchange, not only an exchange, Indodax also provides a chat room containing investors' opinions. Opinions contained in the Indodax chat room can be used to determine whether comments are positive, neutral or negative, so that it can be an investor's decision to sell or buy bitcoin using sentiment analysis. The sentiment analysis process begins with collecting data using an instant data scraper on the Indodax website, data preprcoessing, labeling using vader lexicon, TF-IDF as word weighting, data splitting, naïve Bayes algorithm and support vector machine, feature selection xgboost and gradient boosting, model evaluation with confusion matrix, then comparing the results of the two algorithms. Based on the tests that have been carried out, naïve bayes obtained the best accuracy value of 70.7%, naïve bayes combined with XGBoost obtained the best accuracy value of 86.6%, while the Support vector machine obtained the best accuracy 86.1%, support vector machine combined with gradient boosting obtained the best accuracy value of 88%. Based on these results the use of feature selection can increase the accuracy value of the algorithm.

A. Introduction

Cryptocurrency is a digital currency that uses an encryption system and various forms of this digital currency have spread throughout the world [1]. One of the well-known cryptocurrencies namely bitcoin, bitcoin is a payment network based on peer-to-peer technology and is open source. Every bitcoin transaction is stored in the database of the bitcoin network. When a transaction occurs with bitcoin, buyers and sellers will automatically be recorded in the bitcoin database network [2].

The rapid development of technology makes people want to make payments quickly and easily, bitcoin can be a solution to overcome this. We can do cryptocurrency exchange in Indonesia on the Indodax site. Indodax is one of the well-known cryptocurrency exchanges in Indonesia, not only providing crypto buying and selling services, Indodax also provides chat room services to convey investors' opinions on crypto. Opinions expressed by investors can be used to see what investors think about bitcoin at that time. Opinions that develop can affect the price of bitcoin, therefore it is necessary for investors to know the opinions that are circulating, so that they can determine in making decisions to sell or buy.

Sentiment analysis is the process of understanding and processing textual data automatically to obtain information. Sentiment analysis is performed to detect opinions on a subject and an object (eg individuals, organizations or products) in a data set [3]. Machine learning algorithm models that can be used for sentiment analysis are Support Vector Machine and Naïve Bayes. Support vector machine is a classification method which in its work process uses a hypothetical space consisting of bidirectional linear functions in a high-dimensional feature space. Naïve Bayes is an algorithm that is able to accept input in any form and has speed in processing data. Each new data will be probabilities with each existing class [4]. Naïve Bayes has the disadvantage that it is very sensitive in feature selection [5]. The disadvantages of this algorithm can affect the accuracy value, for that it is necessary to have feature selection. Some examples of feature selection are XGBoost, Gradient Boosting, Adaboost and LightGBM.

Previous research was carried out by [6] Comparing the naive bayes algorithm and the support vector machine for Twitter sentiment analysis, this study obtained the highest accuracy value for the svm algorithm, namely 73.65%. Next study was carried out by [7] comparing 3 algorithms namely naive Bayes, svm and k-nn for analysis of public sentiment towards vaccines, this study obtained the highest accuracy score on the SVM algorithm, namely 83.3%. The last research was carried out by [8] comparing the Naive Bayes and SVM algorithms for sentiment analysis of political figures, this study obtained the highest accuracy score for the SVM algorithm, namely 78.4%. Based on previous studies that did not use feature selection. The novelty in this research is to combine the support vector machine and naive bayes algorithms with feature selection which aims to increase the accuracy value.

B. Research Method

a. Research Stages

Figure 1 presents the steps taken

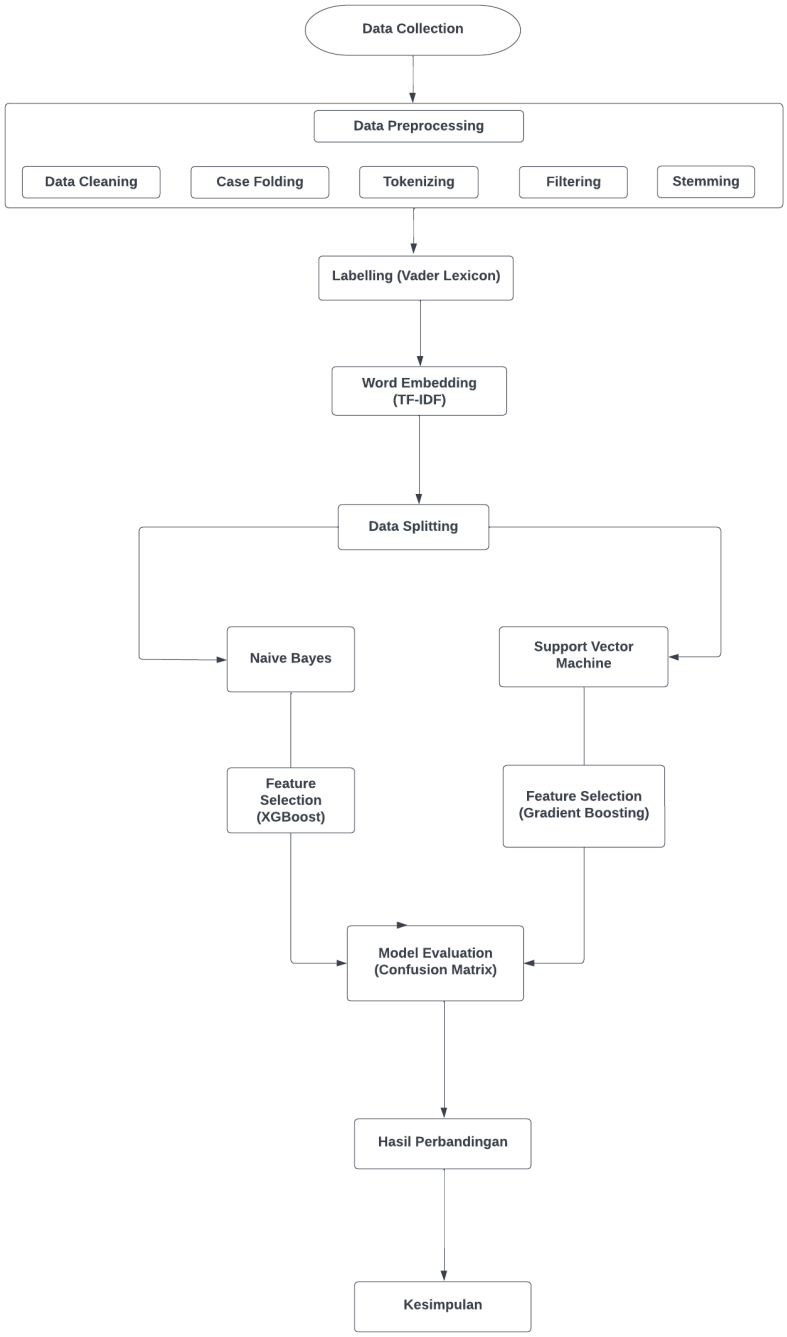

Figure1. Research Flowchart

The process begins with collecting data using a scraping technique using an instant data scraper on the indodax.com website. This data was taken for the period June 26 - July 27 2022. There were 2,890 data samples collected, with 2 variables, namely usernames and comments. The next stage after data collection is to do data preprocessing, this stage aims to remove empty data and clean data from numbers, symbols and links, the steps taken are as follows:

1. Data Cleaning is the stage of filtering data, such as deleting long enough words that are set by a specified limit After scraping[9]
2. Case Folding is the process of equalizing cases in a document. This is done to facilitate the search [10]
3. Tokenizing is a stage of cutting word strings based on the arrangement of these words [11]
4. Filtering is taking important words from tokenizing results or commonly called eliminating words according to the rules [11]
5. Stemming is a very important process to find the base word of a derivative word [12]

After doing preprocessing, then changing the language to English so that the data can be labeled automatically in the labeling process using the vader lexcion, so the labeling is done not manually.

b. Word Embedding (TF-IDF)

The term frequency-inverse document frequency (TF-IDF) word weighting technique is used to gauge a word's significance within a document or text corpus. This approach is predicated on the idea that words that appear more frequently in a given document carry a higher weight than terms that appear more infrequently across a corpus of text. The term "Term Frequency" (TF, for short) describes how frequently a word appears in a document. The TF value increases as the frequency of the term increases. The TF value does not, however, reveal the word's significance within the corpus of the text as a whole.

The IDF (Inverse Document Frequency) statistic calculates the frequency of a word in a corpus of text. Rare terms in the text corpus have a high IDF value, whereas words that are used frequently have a low IDF score. This aids in locating more precise and instructive words. The TF value is multiplied by the IDF value to determine the TF-IDF word weight. This provides terms that are more significant to the document and appear more frequently in the document but less frequently in the overall text corpus more weight.

c. Classification

Data mining techniques like classification are used to determine a data set's class or label based on its features. Building a model that can be used to predict a class of data that has never been seen before is the primary objective of classification. The labeled or classified data is first gathered, and then it is examined for patterns or traits that can be utilized to identify classes in unlabeled or unclassified data. Also, a method that matches the properties of the data observed is used to construct a categorization model.

After built a model, unknown data can be fed into the model after it has been created to forecast its class. By comparing unknown data features with known data features in the training data, this procedure can be carried out. To gauge model performance, there is also a classification model validation technique available.

d. Naïve Bayes

The Naive Bayes algorithm is a classification method based on the Bayes theorem with the straightforward (or "naive") presumption that each feature or characteristic of the data is independent of every other feature or attribute. Based on the information that is known about the properties of the data, this algorithm is used to forecast the category or class of the data. A structured dataset with each instance having a class label and measured attribute is necessary for the Naive Bayes algorithm. The overall class probability is determined by adding the probabilities for each attribute for each class in this algorithm. The following is an explanation of how the Naive Bayes algorithm works:

1. Calculate each class's prior probability, or the likelihood that it will occur in the dataset.
2. Determine the conditional probability of each characteristic for each class, or the likelihood that a particular attribute value will occur in a particular class.
3. Calculate the total probability for each class by adding the prior probabilities and the attribute conditional probabilities.
4. Predict the class that has the highest overall probability using the updated data.

Because the Naive Bayes algorithm can function well on data that has numerous attributes and is particularly effective in training and prediction, it is frequently used in text processing, spam detection, document classification, and image classification.

e. Support Vector Machine

One machine learning method for classifying data and doing regression is the Support Vector Machine (SVM) algorithm. The objective is to choose the optimum line or hyperplane that most effectively divides two different data classes. SVM searches for support vectors that are on the margin edges to find lines/hyperplanes that divide data classes with the largest margins. In order to maximize the margin between distinct data classes, SVM then minimizes the distance between the line/hyperplane and the support vector.

To make data easier to separate, SVM uses kernel algorithms to map it from a low-dimensional space to a higher-dimensional space. SVM can address non-linear issues and avoid overfitting thanks to this kernel approach. SVM provides a number of benefits, including the ability to avoid overfitting issues and having strong performance on data with minimal characteristics. The disadvantage of SVM is that it might be computationally expensive when there is a lot of data and complicated features. The SVM algorithm formula is as follows

$$b = -\frac{1}{2}(W.X^{+} + W.X^{-}) \tag{1}$$

$$W = \sum_{i=1}^{n} a_i \, y_i x_i \tag{2}$$

b = bias value
$W.x^+$ = weight value for positive data class
$W.x^-$ = weight value for negative data class
W = vector weight
a_i = data weight value to i
y_i = data class to i
x_i = data to i

f. Feature Selection

The process of choosing a subset of the characteristics in the data to be used for analysis or prediction in data mining is known as feature selection. The goal of feature selection is to enhance the accuracy of analysis or prediction results by keeping just the pertinent data and removing noise or unimportant features. The benefits of feature selection include improving the interpretation of analytic results, lowering overfitting, boosting prediction accuracy, and saving time and money. It should be highlighted, nonetheless, that improper feature selection might result in the loss of crucial data and lower the standard of analysis or prediction outcomes.

g. Model Evaluation

The confusion matrix is a way to evaluate the performance of a machine learning model used to classify data. The confusion matrix produces a 2x2 matrix that contains information about the correct and incorrect predictions made by the model. Table 2 is an explanation of each element in the confusion matrix:

Table 2. Confusion Matrix

		Prediction
	Positive	Negative
Actual	True Positive	False Negative
	False Positive	True Negative

True Positive (TP): The volume of data that the model properly identified as positive.
False Positive (FP): The quantity of data that the model misclassified as positive.
True Negative (TN): The volume of data that the model accurately identified as negative.
False Negative (FN): The quantity of data that the model misclassified as negative.

C. Result and Discussion

a. Data Collection

The dataset was taken using a scraping technique using the instant data scraper extension, opinions were taken from the indodax.com website chat room, only words containing 'BTC' would be taken, the data collection period was taken within one month, namely from 26 June – 27 july 2022.

After the data is collected, then preprocessing is carried out, then the data is translated into English which aims to do automatic labeling using the vader lexcion, labeling is categorized into three classes namely positive, neutral and negative, the

result is 2890 comments that have been collected. Data will be divided into two parts. Data division will be carried out four times, with a ratio of 60:40, 70:30, 80:20 and 90:10. Data collection results are presented in Figure 2.

	usernames	comments
2885	ryanheart	adedamelaputra82 udah ngejual 14 triliun jual ...
2886	Master_juanda_Sirait	kukuhsetiadi99 thn 2018 1btc 90 juta
2887	PRIHARDIYONO	tinz347 wiiss gede jg ya14 tang jual 14tyang s...
2888	RIZQIWILD	sansanbtc2018 karang kalo idx mt mikir2 notif ...
2889	anwarasri7	btc jatuh

Figure 2. Data Collection

b. Preprocessing

Data that has been collected using a scarping technique with the help of the instant data scraper extension, will then be cleaned, which aims to remove empty data, and clean data from symbols and links. The results of preprocessing are presented in the following table:

1. The result of the data cleaning process are presented in Table 3
2.

Table 3. Data Cleaning Result

Before	After
BTC OTW 17 k cek candle tuh https://ibb.co/YX9spYz	BTC OTW 17 k cek candle tuh

3. The results of the case folding process are presented in Table 4
4.

Table 4. Case Folding Result

Before	After
BTC OTW 17 k cek candle tuh https://ibb.co/YX9spYz	btc otw 17 k cek candle tuh

5. The results of the tokenizing process are presented in Table 5
6.

Table 5. Tokenizing Result

Before	After
BTC OTW 17 k cek candle tuh	[btc, otw, 17, k, cek, candle, tuh]

7. The results of the filtering process are presented in Table 6

Table 6. Filtering Result

Before	After
menurutku ini btc masih break dan jangan lupa double bottom tf 1d masih	['menurutku', 'btc', 'break', 'lupa', 'double', 'bottom', 'tf', '1d', 'turun']

8. The results of the stemming process are presented in Table 7

Table 7. Stemming Result

Before	After
menurutku ini btc masih break dan jangan lupa double bottom tf 1d masih	turut btc break lupa double bottom tf 1d turun

c. Word Embedding TF-IDF

Term frequency–inverse document frequency or abbreviated as TF-IDF is used to count the occurrence of words in each data. The words will be converted into numeric form so that they can be processed by a computer. The calculation results are presented in Table 8

Table 8. TF-IDF Value

Word	Value
resist	0.458485
why	0.458485
down	0.451522
do	0.357442
you	0.290116
already	0.241223

d. Naïve Bayes with XGBoost

The data will be processed using the Naïve Bayes algorithm model, to improve the performance of the Naïve Bayes model, it will be combined with XGBoost, the processing will be carried out using the Python programming language. The test results with the Naive Bayes algorithm are presented in table 9

Table 9. Naive Bayes Algorithm Accuracy Value

Data Splitting	Naïve Bayes Accuracy
60:40	69.6%
70:30	70.7%
80:20	70.2%
90:10	68.1%

Based on Table 9, it can be seen that the accuracy value of the naïve Bayes algorithm is not satisfactory, for this reason feature selection is needed in order to increase the accuracy value. This study uses XGBoost to improve the performance of the naïve Bayes algorithm model, the use of XGBoost is presented in Figure 3.

```
In [16]: from xgboost import XGBClassifier

In [17]: classifier = MultinomialNB()
         xgbc = XGBClassifier(classifier,10)

         xgbc.fit(X_train, y_train)

         pred_xgboost = xgbc.predict(X_test)
```

Figure 3. XGBoost

After adding XGBoost it was found that the accuracy value increased, the increase in accuracy is presented in Table 10

Table 10. Naive Bayes Algorithm with XGBoost Accuracy Value

Data Splitting	Naïve Bayes Accuracy	Naïve Bayes + XGBoost Accuracy
60:40	69.6%	82.7%
70:30	70.7%	85.2%
80:20	70.2%	86.6%
90:10	68.1%	85.4%

Based on Table 11, the highest accuracy value after adding feature selection XGBoost feature is found in the 80:20 data splitting, which is 86.6%

e. Support Vector Machine with Gradient Boosting

In addition to processing data using the naïve Bayes algorithm, this research also processes data using a support vector machine algorithm combined with a gradient boosting feature selection, the aim is to compare the algorithms that have the highest scores. After being processed with the support vector machine algorithm, an evaluation of the model is carried out to find the accuracy value of each data splitting. Based on Table 11 it can be seen that the accuracy value of each data splitting, for data splitting on the support vector machine algorithm

Table 11. Support Vector Machine Algorithm Accuracy Value

Data Splitting	Support Vector Machine Accuracy
60:40	82.9%
70:30	84.6%
80:20	85.2%
90:10	86.1%

Based on Table 11 it can be seen that the accuracy value of the support vector machine algorithm, the accuracy obtained can be added with feature selection to increase the accuracy value, this study uses Gradient Boosting to improve the performance of the support vector machine algorithm model, the use Gradient Boosting is presented in Figure 4

```
In [50]: from sklearn.ensemble import GradientBoostingClassifier

gbc_clf = GradientBoostingClassifier( max_depth=15, random_state=42)
gbc_clf.fit(X_train, y_train)

pred_gbt = gbc_clf.predict(X_test)
```

Figure 4. Gradient Boosting

After adding Gradient Boosting, it was found that the accuracy value increased, the increase in accuracy is presented in Table 12

Table 12. Support Vector Machine Algorithm with Gradient Boosting Accuracy Value

Data Splitting	Support Vector Machine Accuracy	Support Vector machine + Gradient Boosting Accuracy
60:40	82.9%	84.6%
70:30	84.6%	86.6%
80:20	85.2%	87.7%
90:10	86.1%	88%

Based on Table 12, the highest accuracy value after adding feature selection gradient boosting feature is found in the 90:10 data splitting, which is 88%

f. Model Evaluation

Confusion matrix is a table used to measure the performance of a machine learning model by comparing the model's predicted results with actual values. Four metrics—True Positive, False Positive, True Negative, and False Negative—are presented in the table. The model's accuracy, precision, recall, and F1-score are determined using these criteria. The confusion matrix is crucial for assessing model performance and aids machine learning developers in comprehending the effectiveness of their models. The results of the confusion matrix for the highest score on the SVM algorithm combined with Gradient Boosting are presented in Figure 5

```
In [20]: #Hasil Performa model Gradient Boosting
         from sklearn.metrics import classification_report

         print(classification_report(pred_gbt,y_test, zero_division=0))
```

```
              precision    recall  f1-score   support

    Negative       0.58      0.93      0.71        27
     Neutral       0.98      0.87      0.92       208
    Positive       0.79      0.91      0.84        54

    accuracy                           0.88       289
   macro avg       0.78      0.90      0.83       289
weighted avg       0.91      0.88      0.89       289
```

Figure 5. Confusion Matrix Results

Based on figure 5, the value of this research gets the highest score, namely accuracy 88%, precision 78%, recall 90% and f1-score 83%.

g. Comparison Results

After testing with two naïve Bayes algorithm models and a support vector added with feature selection, the final result is to compare which algorithm has the highest level of accuracy for each splitting data. The results of the comparison are presented in Table 13

Table 13. Comparison Results

Data Splitting	Naïve Bayes	Naïve bayes + XGBoost	SVM	SVM + Gradient Boosting
60:40	69.6%	82.7%	82.9%	84.6%
70:30	70.7%	85.2%	84.6%	86.6%
80:20	70.2%	86.6%	85.2%	87.7%
90:10	68.1%	85.4%	86.1%	87.8%

Based on the comparison results for the two algorithms which are both added with feature selection, in the Naive Bayes algorithm the highest accuracy value is found in the 80:20 data splitting, which is 86.6%, while in the SVM algorithm, the highest accuracy value is found in the 90:10 data splitting, which is 88%.

D. Conclusion

Based on the results of the study, 2,890 comments were taken on the indodax site. The data has been tested with the naïve Bayes algorithm and support vector machine, each of which is added with feature selection. It can be concluded that adding feature selection can increase the accuracy value of the machine learning algorithm. The support vector machine algorithm combined with gradient boosting on the 90:10 data splitting obtains the highest accuracy values of 88%, 78% precision, 90% recall and 83% f1-score. By adding feature selection, we can increase the accuracy value of the model created. The researcher provides suggestions for future researchers to be able to increase the accuracy value by trying other algorithms in the classification model.

E. Acknowledgment

The researcher suggests thanks to STMIK Amik Riau and supervisors who have helped in this research.

F. References

[1] L. Scientia, "16 https://journal.unnes.ac.id/sju/index.php/lslr/ UKM Lex Scientia, Fakultas Hukum Universitas Negeri Semarang," 2019. [Online]. Available: https://journal.unnes.ac.id/sju/index.php/lslr/

[2] M. Masruron Dan, M. Al Azhari, L. Timur, and N. Tenggara Barat, "TINJAUAN HUKUM ISLAM TERHADAP TRANSAKSI BITCOIN DALAM PERSPEKTIF ULAMA FIQH KLASIK DAN KONTEMPORER." [Online]. Available: http://bitcoin.org/id,

[3] C. F. Hasri and D. Alita, "PENERAPAN METODE NAÏVE BAYES CLASSIFIER DAN SUPPORT VECTOR MACHINE PADA ANALISIS SENTIMEN TERHADAP DAMPAK VIRUS CORONA DI TWITTER," *Jurnal Informatika dan Rekayasa Perangkat Lunak (JATIKA)*, vol. 3, no. 2, pp. 145–160, 2022, [Online]. Available: http://jim.teknokrat.ac.id/index.php/informatika

[4] M. Ridho Handoko, "SISTEM PAKAR DIAGNOSA PENYAKIT SELAMA KEHAMILAN MENGGUNAKAN METODE NAIVE BAYES BERBASIS WEB,"

Jurnal Teknologi dan Sistem Informasi (JTSI), vol. 2, no. 1, pp. 50–58, 2021, [Online]. Available: http://jim.teknokrat.ac.id/index.php/JTSI

[5] H. Muhabatin *et al.*, "Klasifikasi Berita Hoax Menggunakan Algoritma Naïve Bayes Berbasis PSO," *Informatics for Educators and Professionals*, vol. 5, no. 2, pp. 156–165, 2021, [Online]. Available: https://turnbackhoax.id/

[6] M. I. Fikri, T. S. Sabrila, Y. Azhar, and U. M. Malang, "Perbandingan Metode Naïve Bayes dan Support Vector Machine pada Analisis Sentimen Twitter".

[7] R. T. Aldisa and P. Maulana, "Analisis Sentimen Opini Masyarakat Terhadap Vaksinasi Booster COVID-19 Dengan Perbandingan Metode Naive Bayes, Decision Tree dan SVM," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 1, pp. 106–109, Jun. 2022, doi: 10.47065/bits.v4i1.1581.

[8] S. Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti *et al.*, "Terakreditasi SINTA Peringkat 2 Perbandingan Metode Klasifikasi Analisis Sentimen Tokoh Politik Pada Komentar Media Berita Online," *masa berlaku mulai*, vol. 1, no. 3, pp. 176–183, 2017.

[9] M. E.-K. Kesuma and R. Iskandar, "Analisis Toko dan Asal Toko Fashion Pria di Shopee Menggunakan Data Scrapping dan Exploratory Data Analysis," *Majalah Ilmiah Teknologi Elektro*, vol. 21, no. 1, p. 127, Jul. 2022, doi: 10.24843/mite.2022.v21i01.p17.

[10] D. Alita and A. Rahman, "Pendeteksian Sarkasme pada Proses Analisis Sentimen Menggunakan Random Forest Classifier," 2020.

[11] A. Witanti, B. Yogyakarta Jl Raya Wates-Jogjakarta, K. Sedayu, K. Bantul, and D. Istimewa Yogyakartalamat, "ANALISIS SENTIMEN MASYARAKAT TERHADAP VAKSINASI COVID-19 PADA MEDIA SOSIAL TWITTER MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE (SVM)," *Jurnal Sistem Informasi dan Informatika (Simika) P-ISSN*, vol. 5, pp. 2622–6901, 2022.

[12] A. Guterres, Gunawan, and J. Santoso, "Stemming Bahasa Tetun Menggunakan Pendekatan Rule Based," *Teknika*, vol. 8, no. 2, pp. 142–147, Oct. 2019, doi: 10.34148/teknika.v8i2.224.