Scientific Journal of Informatics
Vol. 9, No. 1, May 2022
p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

Implementation of Stacking Ensemble Classifier for Multi-class Classification of COVID-19 Vaccines Topics on Twitter

Rama Jayapermana[1*], Aradea[2], Neng Ika Kurniati[3]
[1,2,3]Informatics Department, Faculty of Engineering, Universitas Siliwangi, Indonesia

Abstract.
Purpose: However, from the variety of uses of these algorithms, in general, accuracy problems are still a concern today, even accuracy problems related to multi-class classification still require further research.
Methods: This study proposes a stacking ensemble classifier method to produce better accuracy by combining Logistic Regression, Random Forest, and Support Vector Machine (SVM) algorithms as first-level learners and using Logistic Regression as a meta-learner for the multi-class classification of COVID-19 vaccine topics on Twitter.
Result: Based on the evaluation, the proposed Stacking Ensemble Classifier model shows 86% accuracy, 85% precision, 86% recall, and 85% f1-score.
Novelty: The novelty is produce better accuracy by combining Logistic Regression, Random Forest, and Support Vector Machine (SVM) algorithms as first-level learners and using Logistic Regression as a meta-learner.

Keywords: COVID-19 Vaccines, Ensemble Method, Multi-class Classification, Sentiment Analysis, Stacking Ensemble Classifier
Received August 2021 / **Revised** November 2021 / **Accepted** February 2022

INTRODUCTION

Vaccines are a further effort to tackle the COVID-19 virus that has spread to various regions in Indonesia. The government's move to hold a COVID-19 vaccination program is the right thing. However, sundry public opinions about this program emerged, ranging from positive to negative opinions. One of the most talked about is the first vaccination against President Joko Widodo on January 13, 2021, at the State Palace [1]. Social media has become the most popular place for a person to express his opinion on various things. That matter happens because social media is something that everyone should have [2]. One of the most widely used social media in Indonesia is Twitter. The use of social media Twitter in Indonesia reaches 63.6% of internet users aged 16 to 64 years [3]. That matter opens up many sources of data that can use to generate knowledge. Twitter is also one of the social media that is often used by researchers in collecting data to support their research. One area of research that often uses data from Twitter is sentiment analysis.

Sentiment analysis or also called opinion mining is one of the studies on text mining that aims to determine public perception or subjectivity to a topic, event, or problem [4]. Several studies analyze public opinion regarding the topic of COVID-19 on Twitter. Such as research [5] which classifies sentiments related to the PSBB topic on Twitter social media by comparing the Support Vector Machine (SVM) algorithm with Random Forest. The results show an accuracy of 58% for the Random Forest algorithm and show an accuracy of 56% for the Support Vector Machine (SVM) algorithm. The results of this study are very low due to the use of very little data, which is only 499 data.

Research [6] conducted a classification of Twitter social media users' perceptions of COVID-19 cases using the Logistic Regression algorithm with “L2” and “None” hyperparameters using 44955 tweet data. The results of this study indicate that the Logistic Regression algorithm with “L2” parameters produces an accuracy value of 77% with an f1-score of 74%, while the None parameter produces an accuracy value of 74% with an f1-score of 70%. The use of large data in this study can affect the accuracy of the Logistic

*Corresponding author.
Email addresses: 177006099@student.unsil.ac.id (Jayapermana), aradea@unsil.ac.id (Aradea), nengikakurniati@unsil.ac.id (Kurniati)
DOI: 10.15294/sji.v9i1.31648

Regression algorithm. This is because some algorithms such as the Support Vector Machine (SVM) are not efficient in training large-capacity data [7].

In other topics, research [7] conducted a web content analysis for fake news using the Max Voting and Stacking Ensemble Classifier methods. As a result, the combination of the Logistic Regression, Support Vector Machine, and Random Forest algorithms in the Stacking Ensemble Classifier method got the best performance with an accuracy value of 98.37%, precision 0.97, recall 0.99, f1-score 0.98, and AUC 0.9888. Meanwhile, in dataset 2, the accuracy is 94.32%, precision is 0.95, recall is 0.93, f1-score is 0.94, and AUC is 0.9853. This performance is better than using the Max Voting ensemble which got an accuracy of 97.89%, precision 0.97, recall 0.99, f1-score 0.98, and AUC 0.9683 for dataset 1, as well as the accuracy of 93.85%, precision 0.94, recall 0.90, f1-score 0.92, and AUC 0.9625 for dataset 2. Research [8] uses a stacked ensemble method with Naive Bayes algorithm and Support Vector Machine to detect sentiment polarity in Bengali tweets with negative, neutral, and positive sentiment. As a result, the proposed stacked ensemble model achieves an accuracy of 56.2% which exceeds the LSTM algorithm which obtains an accuracy of 55.27%.

Research [9] optimized sentiment analysis for the 2019 presidential election with the Naive Bayes algorithm and Particle Swarm Optimization (PSO). The result of the accuracy with the 10-fold cross-validation for Naive Bayes algorithm of 86.62% and Naive Bayes with Particle Swarm Optimization (PSO) of 90.74%. The accuracy increases by 4.12% for Naive Bayes with Particle Swarm Optimization (PSO) algorithm. Research [10] proposed a new method with Weighted Classifier Selection and Stacked Ensemble for multi-label classification, resulting in an accuracy value of 0.909 for the MLWSE-L1 model and 0.91 for the MLWSE-L2 model. Research [11] conducted a sentiment analysis on the COVID-19 vaccine in India, the results concluded that the Covaxin vaccine had 36% positive tweets and 31% negative tweets, while the Covishield vaccine had 71% positive tweets and 29% negative tweets.

Despite achieving an accuracy of up to 98%, the study [7] only dealt with binary-class classification. Meanwhile, studies that perform multi-class classifications such as research [5], [6], and [8] get a low accuracy value. In this study, we combined the Logistic Regression algorithm, Support Vector Machine (SVM), and Random Forest as first-level learners, then used Logistic Regression as a meta-learner in the Stacking Ensemble Classifier method to improve the accuracy of multi-class classification of the COVID-19 vaccine topic on Twitter.

METHODS

The purpose system in Figure 1 is a process of our system. There as six main stages of the process, i.e., preprocessing, dataset labeling, feature extraction using TF-IDF, splitting the data, modeling, and evaluation.

Data

The data used in this study is tweet data related to the topic of the COVID-19 vaccine written by Indonesian people on Twitter. The data was collected using the *twint* library [12] in the python programming language. Data was collected using the keyword "*vaksin covid19*". The tweet data is tweet data available from May 1, 2021, to July 10, 2021. The data that has been collected is 13011 tweets. Table 1 shows a sample of tweet data that has been collected.

Table 1. Sample of tweet data

Tweet
@EnggalPMT @imbangimedia Bahaya Nih... Ternyata MODUS BELAKA... Rakyat Indonesia Jadi KELINCI PERCOBAAN VAKSIN Covid19.
Ini covid sebelum vaksinasi massal digaungkan oleh pemerintah kok kasus udah jarang kita dapat pemberitaan orang meninggal. Propaganda agar masyarakat ketakutan pun sempat sangat minim Setelah program vaksin massal besar-besaran kok malah melunjak lagi yah kasusnya #COVID19
Syukur Alhamdulillah, setelah 3 bulan menunggu, akhirnya dapat juga tarikh suntikan vaksin covid-19. Jom ke Shah Alam kita. #vaksìncovid19 #pfizer #lindungdirilindungsemua @ Taman Desa Kesang https://t.co/OjKn1Wtlvc
Dukung Pelaksanaan Vaksinasi Covid-19 Untuk Percepat Pemulihan Ekonomi Indonesia #vaksin #sinovachalal #vaksìncovid19 #vaksingratisutkrakyat #covid19indonesia #Covid19 #lfl #likeforlikes #Jokowi #Indonesia #IndonesiaMaju https://t.co/AMWzn5CT2P

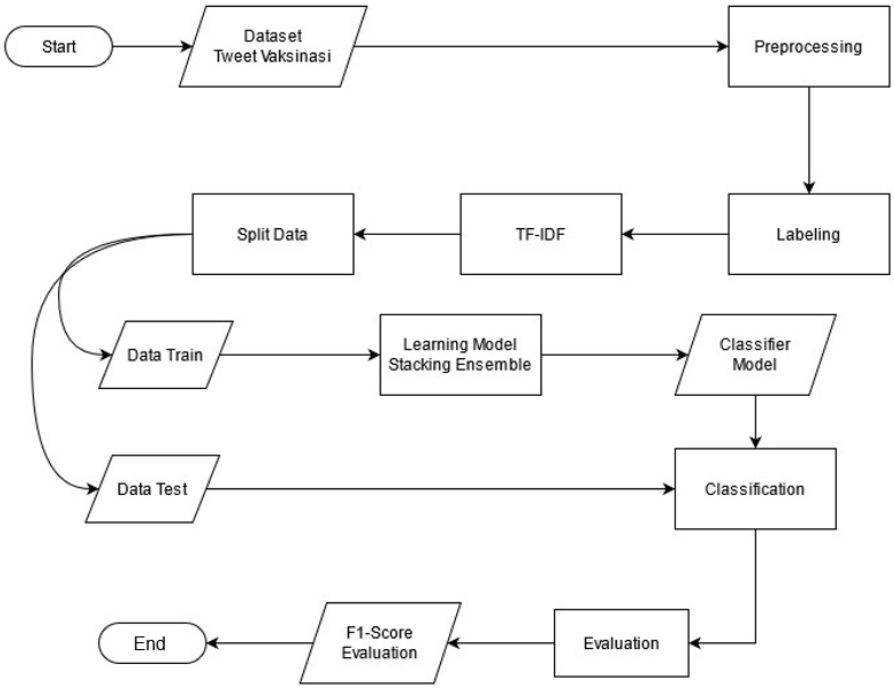

Figure 1. Purposed system

Preprocessing

In text mining, the preprocessing stage is carried out to process various data into regular data that can be applied to machine learning algorithms [13]. Figure 2 shows the preprocessing steps in this study: cleansing, case-folding, normalization, tokenizing, removing stop words, and stemming.

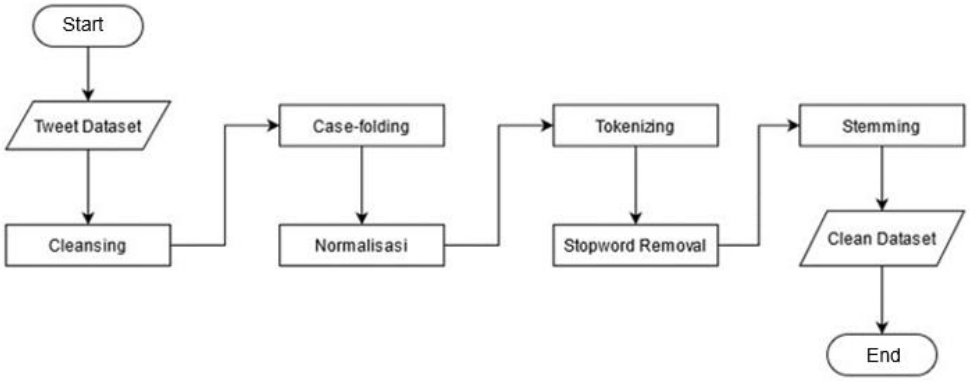

Figure 2. Preprocessing steps

The first step in the preprocessing stage is cleansing which functions to clean tweet data from unnecessary elements, such as hashtags starting with the # symbol, mentions starting with the @ symbol, links, retweets starting with RT, punctuation marks, numbers, images, and whitespace. The case-folding stage is used to convert uppercase letters to lowercase letters. The normalization stage is used to change short and "*alay*" words, into normal words by using "*kamus alay*" [14]. The tokenizing stage uses the *nltk* library [15] to break the sentence into a collection of words that compose the sentence. The stopword removal step is used to remove prepositions, pronouns, and conjunctions. The stemming stage uses the *PySastrawi* library [16] to change words into basic words to reduce word dimensions [17].

Dataset Labeling

The data is labeled positive, negative, or neutral at this stage. In this study, data labeling uses the InSet (Indonesian Sentiment Lexicon) dictionary [18] to calculate the polarity of tweet data. In the InSet

dictionary, there are different polarities for each word, both positive and negative words. A negative number represents the polarity for negative words, while a positive number describes the polarity for positive words. The polarity of the tweet is calculated for each word in the sentence. The total number of polarities will determine the polarity of the sentence. If the total polarity is 0, then the tweet is labeled neutral. If the total polarity is less than 0, then the tweet is labeled negative. If the total polarity is more than 0, then the tweet is labeled positive.

Feature Extraction Using TF-IDF

This research uses the TF – IDF method with the *sklearn* library [19]. Term Frequency–Inverse Document Frequency (TF-IDF) is a numerical statistic to indicate the importance of a word in a document from a list of documents or corpus [20]. Term Frequency (TF) is the number of times a word occurs in a document which is defined as [20]:

$$TF_{t,d} = \frac{f_{t,d}}{\text{number of words in d}} \quad (1)$$

where $f_{t,d}$ is the number of occurrences of the word t in document d. Inverse Document Frequency (IDF) is a measure of how much information a word provides [19]. Unlike the traditional IDF, the IDF on *sklearn* uses a unitary constant in the denominator and numerator which is defined as [19]:

$$idf(t) = \log \frac{1+n}{1+df(t)} + 1 \quad (2)$$

where n is the number of documents in the corpus and $df(t)$ the number of occurrences of the word t in all documents in the corpus.

Split Data

Based on the feature extraction results using TF – IDF, the data is divided into two parts with 80:20 division. Data with 80% is used as training data, and 20% is used as test data.

Modeling

In this study, we propose a model using the stacking ensemble classifier method by combining the Logistic Regression algorithm, Support Vector Machine (SVM), and Random Forest as first-level learners and using the Logistic Regression algorithm for meta-learners. Figure 3 shows the process flow in the proposed model.

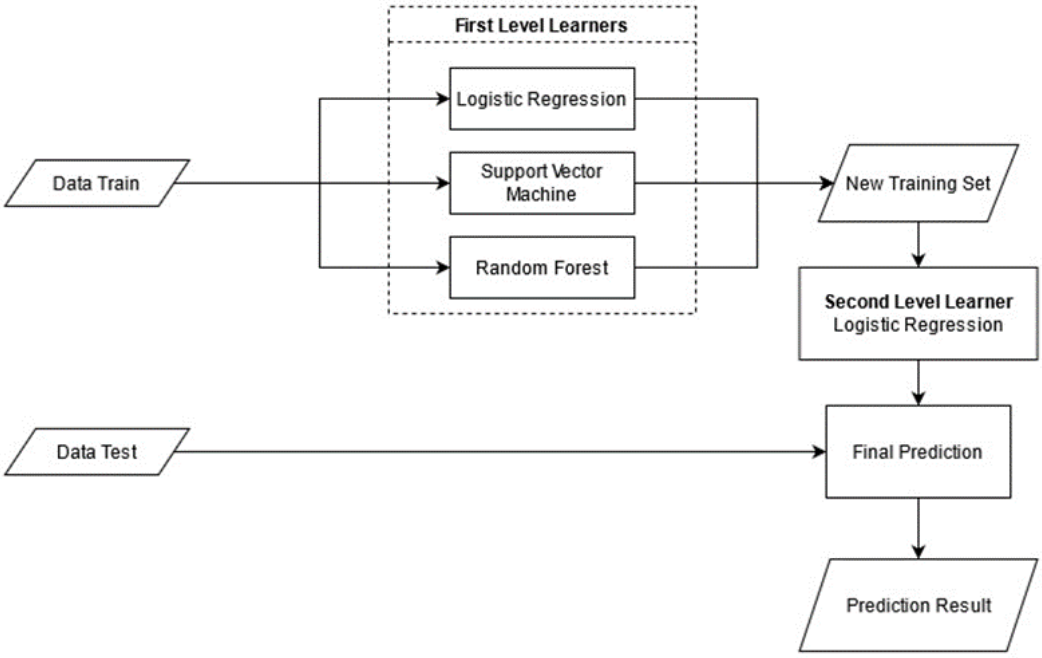

Figure 3. Process flow of proposed model

Evaluation

Evaluation of the model made in this study uses a confusion matrix to calculate the values of accuracy, precision, recall, and f1-score. Table 2 is a confusion matrix table in this study for the case of multi-class classification.

Table 2. Confusion matrix for multi-class classification [21]

		Predicted		
		Positive	**Neutral**	**Negative**
Actual	**Positive**	P_{PP}	P_{NtP}	P_{NgP}
	Neutral	P_{PNt}	P_{NtNt}	P_{NgNt}
	Negative	P_{PNg}	P_{NtNg}	P_{NgNg}

Based on Table 2, we can calculate accuracy, precision, and recall as:

$$Accuracy = \frac{P_{PP}+NgNg+NtNt}{P_{PP}+NgNg+\cdots+P_{PNg}+P_{NtNg}} \; x\; 100\% \tag{3}$$

$$Prec.Pst = \frac{P_{PP}}{P_{PP}+P_{PNt}+P_{PNg}} \; x\; 100\% \tag{4}$$

$$Prec.Ng = \frac{P_{NgNg}}{P_{NgNg}+P_{NgNt}+P_{NgP}} \; x\; 100\% \tag{5}$$

$$Prec.Nt = \frac{P_{NtNt}}{P_{NtNt}+P_{NtP}+P_{NtNg}} \; x\; 100\% \tag{6}$$

$$Rec.Pst = \frac{P_{PP}}{P_{PP}+P_{NtP}+P_{NgP}} \; x\; 100\% \tag{7}$$

$$Rec.Ng = \frac{P_{NgNg}}{P_{NgNg}+P_{NtNg}+P_{PNg}} \; x\; 100\% \tag{8}$$

$$Rec.Nt = \frac{P_{NtNt}}{P_{NtNt}+P_{PNt}+P_{NgNt}} \; x\; 100\% \tag{9}$$

Then f1-score as:

$$F1-Score.pos = 2\; x\; \frac{prec.Pst\; x\; rec.Pst}{pre.Pst + rec.Pst} \tag{10}$$

$$F1-Score.neg = 2\; x\; \frac{prec.Ng\; x\; rec.Ng}{prec.Ng + rec.Ng} \tag{11}$$

$$F1-Score.nt = 2\; x\; \frac{prec.Nt\; x\; rec.Nt}{prec.Nt + rec.Nt} \tag{12}$$

RESULT AND DISCUSSION

Implementation

The proposed system is implemented in the python programming language with the *nltk* and *PySastrawi* libraries in the preprocessing stage, as well as the *StackingClassifier* module in the *sklearn* library for the implementation of the proposed model.

Testing

Testing is done by training the model using data as much as 80% of the total data or about 9120 data. The time required for this model training process is about 58 seconds. Then predictions were made using test data as much as 20% of the total data or about 2280 data. The prediction results are compared with the actual data and loaded into the confusion matrix. Table 3 is the result of the confusion matrix on the proposed model.

Table 3. Result of the confusion matrix on the proposed model

		Predicted		
		Positive	**Neutral**	**Negative**
Actual	**Positive**	1137	25	69
	Neutral	79	144	53
	Negative	70	27	676

In Table 3, the prediction results for the negative class are 1137 data classified as a negative class, 25 data classified as a neutral class, and 69 data classified as the positive class. The prediction result for the neutral class is 144 data classified as a neutral class, 79 data classified as a negative class, and 53 data classified as the positive class. The prediction result for the positive class is 676 data classified as a positive class, 15 data classified as a neutral class, and 70 data classified as negative class. Based on the confusion matrix in Table 3, the accuracy, precision, recall, and f1-score values can be calculated, the results are shown in Figure 4.

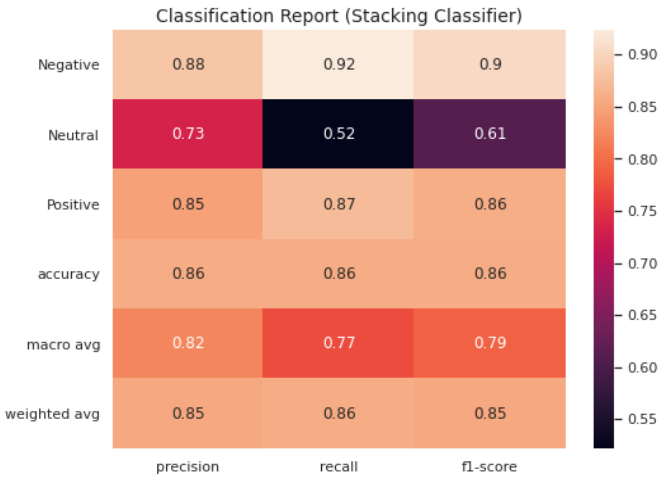

Figure 4. Classification report of purposed model

The results of the classification report in Figure 4 show that the stacking ensemble classifier model in this study has a good performance. That matter is because the accuracy value obtained in this stacking ensemble classifier model is 0.86. This is high enough for a classification model that handles three classes. The precision value for the negative class got 0.88, the neutral class got 0.73, and the positive class got 0.85, with the average precision of each class using the weighted method got 0.85. The recall value for the negative class got 0.92, the neutral class got 0.52, and the positive class got 0.87, with the average recall value of each class using the weighted method got 0.86. The f1-score value for the negative class got 0.9, the neutral class got 0.61, and the positive class got 0.86, with an average f1-score value for each class using the weighted method got 0.86.

Evaluation

Based on the results of the tests carried out, the proposed stacking ensemble classifier model has a better performance compared to other models with a similar case study, namely multi-class classification, the comparison is shown in Table 4.

Table 4. Model performance comparison

Model	**Accuracy (%)**	**F1-Score (%)**
LR + L2 [6]	77	77
Random Forest [5]	58	44
SVM [5]	56	44
Stacking (LR + SVM + RF)	**86**	**85**

In Table 4, it can be seen that the model proposed in this study has a better level of accuracy in handling multi-class classification. Compared to the Logistic Regression + L2 model which uses more data, our proposed model uses fewer data. Our model is 9% better on accuracy and 8% better on the f1-score. Our proposed model is also better than the Random Forest and Support Vector Machine models. With the use of more data, our proposed model is 28% better on the accuracy and 41% on the f1-score against the Random Forest model. Compared with Support Vector Machine model, our model is also 30% better on the accuracy and 41% on the f1-score.

Threats to Validity

We limit the text features generated from the TF-IDF process to 5000 features. The goal is to minimize unnecessary text features. As a result, the accuracy obtained is quite good, reaching 86%. We realize that under other situations if the feature is not limited or reduced, it will affect the accuracy results obtained. The results of the accuracy can be greater or less.

The composition of the data used in this study is also unbalanced data. That matter can also affect the accuracy of results obtained when using balanced data. In addition, we also use the help of a dictionary in data labeling so that there are labels that do not match the sentences in the tweet data. For example, a positive sentence should be labeled negative. That matter will be very different if create labels manually, which will create labels according to the sentences in the tweet data.

CONCLUSION

The proposed stacking ensemble classifier model is a model that implements the stacking ensemble method using the Logistic Regression algorithm, Support Vector Machine (SVM), and Random Forest as first-level learners. Then we uses the Logistic Regression algorithm as a meta-learner which aims to handle multi-level classification. class. This model produces better accuracy and f1-score values than other similar case studies. Compared to the Random Forest + L2 model which uses 44955 tweets, although it uses less data, the proposed model has a 9% better accuracy value and an 8% better f1-score value. This is because the proposed model combines algorithms to reduce bias, more precisely, the predictions of each estimator are stacked together and used as input for the final estimator to calculate predictions. In addition, the proposed model is also compared with the model using the Random Forest and Support Vector Machine (SVM) algorithm for multi-class classification. The proposed model is 28% better on the accuracy value and 41% more prolific on the f1-score value compared to the Random Forest model. Meanwhile, compared to the SVM model, the proposed model excels by 30% on the accuracy value and 41% on the f1-score value. This is because more data was used in this study, so the model gained more knowledge during the training process. This study limits the text features generated from the TF-IDF process to 5000 features. The dataset used in this study is also unbalanced. Apart from that, we also use the help of a dictionary to label the datasets. These things can affect the accuracy results obtained by the proposed model if it is different from what was done in this study.

REFERENCES

[1] CNN Indonesia, "Jokowi Terima Suntikan Dosis Pertama Vaksin Covid-19 Sinovac," 2021. https://www.cnnindonesia.com/nasional/20210112211001-20-592885/jokowi-terima-suntikan-dosis-pertama-vaksin-covid-19-sinovac (accessed Mar. 18, 2021).

[2] F. F. Rachman and S. Pramana, "Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter," *Heal. Inf. Manag. J. ISSN*, vol. 8, no. 2, pp. 2655–9129, 2020.

[3] S. Kemp, "Digital 2020: Indonesia," 2020. https://datareportal.com/reports/digital-2020-indonesia (accessed Mar. 18, 2021).

[4] S. Pramana, B. Yuniarto, S. Mariyah, I. Santoso, and R. Nooraeni, *Data mining dengan R : konsep serta implementasi*. Bogor: In Media, 2018.

[5] M. R. Adrian, M. P. Putra, M. H. Rafialdy, and N. A. Rakhmawati, "Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB," *J. Inform. Upgris*, vol. 7, no. 1, Jun. 2021.

[6] A. K. Santoso, A. Noviriandini, A. Kurniasih, B. D. Wicaksono, and A. Nuryanto, "Klasifikasi Persepsi Pengguna Twitter Terhadap Kasus Covid-19 Menggunakan Metode Logistic Regression," *J. Inform. dan Komput.*, vol. 5, no. 2, pp. 234–241, 2021.

[7] P. Meel, P. Chawla, S. Jain, and U. Rai, "Web Text Content Credibility Analysis using Max Voting

and Stacking Ensemble Classifiers," in *2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA)*, Jul. 2020, pp. 157–161.
[8] K. Sarkar, "A Stacked Ensemble Approach to Bengali Sentiment Analysis," in *Intelligent Human Computer Interaction*, 2020, pp. 102–111.
[9] N. Hayatin, G. I. Marthasari, and L. Nuarini, "Optimization of Sentiment Analysis for Indonesian Presidential Election using Naïve Bayes and Particle Swarm Optimization," *J. Online Inform.*, vol. 5, no. 1, 2020.
[10] Y. Xia, K. Chen, and Y. Yang, "Multi-label classification with weighted classifier selection and stacked ensemble," *Inf. Sci. (Ny).*, vol. 557, pp. 421–442, 2021.
[11] A. D. Dubey, "Public Sentiment Analysis of COVID-19 Vaccination Drive in India," *SSRN Electron. J.*, 2021.
[12] C. Zacharias and F. Poldi, "TWINT - Twitter Intelligence Tool," 2020. https://github.com/twintproject/twint.
[13] A. T. J. Harjanta, "Preprocessing Text untuk Meminimalisir Kata yang Tidak Berarti dalam Proses Text Mining," *Inform. UPGRIS*, vol. 1, pp. 1–9, 2015.
[14] N. A. Salsabila, Y. A. Winatmoko, A. A. Septiandri, and A. Jamal, "Colloquial Indonesian Lexicon," in *2018 International Conference on Asian Language Processing (IALP)*, Nov. 2018, pp. 226–229.
[15] NLTK Project, "Natural Language Toolkit," 2021. https://www.nltk.org/ (accessed Jun. 25, 2021).
[16] H. A. Robbani, "Sastrawi Python," 2016. https://github.com/har07/PySastrawi (accessed Jun. 25, 2021).
[17] I. Najiyah and I. Hariyanti, "Sentimen Analisis Covid-19 Dengan Metode Probabilistic Neural Network Dan Tf-Idf," *J. Responsif Ris. Sains dan Inform.*, vol. 3, no. 1, pp. 100–111, 2021.
[18] F. Koto and G. Y. Rahmaningtyas, "Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs," in *2017 International Conference on Asian Language Processing (IALP)*, Dec. 2017, pp. 391–394.
[19] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
[20] A. Rajaraman and J. D. Ullman, "Data Mining," in *Mining of Massive Datasets*, Cambridge: Cambridge University Press, 2011, pp. 1–17.
[21] C. H. Yutika and S. Al Faraby, "Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes," *J. Media Inform. Budidarma*, vol. 5, pp. 422–430, 2021.