DOI: 10.15849/IJASCA.240330.14

Int. J. Advance Soft Compu. Appl, Vol. 16, No. 1, March 2024
Print ISSN: 2710-1274, Online ISSN: 2074-8523

COVID-19 Vaccination Sentiment Analysis in Indonesia

A Data-Driven Approach

Liana and Seng Hansun

Informatics Department, Universitas Multimedia Nusantara, Indonesia
e-mail: liana1@student.umn.ac.id
Informatics Department, Universitas Multimedia Nusantara, Indonesia
e-mail: seng.hansun@lecturer.umn.ac.id

Abstract

Although it has been lingering for more than three years, COVID-19 still become a challenging health issue nowadays. COVID-19 vaccination was introduced as one strategic approach to managing the COVID-19 pandemic. However, the community response to the solution varied from one to another, especially during the first year of the COVID-19 vaccination program. In this research, we conduct a sentiment analysis to classify whether an expression shared on the Twitter platform is a positive, neutral, or negative sentiment on the COVID-19 vaccination issue, specifically in Indonesia. The classification model used in this research is the Multinomial Naïve Bayes algorithm with several term weighting models, such as Term Frequency-Inverse Document Frequency (TF-IDF) Vectorizer and Count Vectorizer. From the experimental result that was done, poor performances in terms of accuracy and F1-score were found on the imbalanced dataset. Meanwhile, increased performances were detected when the same model was performed on a balanced dataset. The best classification was obtained using TF-IDF Vectorizer with an F1-score of 82.81% and an accuracy of 83.18%.

Keywords: *COVID-19, Multinomial Naïve Bayes, sentiment analysis, TF-IDF vectorizer, Twitter, vaccination.*

1 Introduction

At the end of 2019, towards the beginning of 2020, the whole world received information about a new virus, COVID-19. This virus is contagious and a serious disease, hence it was declared a pandemic in March 2020 by the World Health Organization (WHO) [1]. The cure and the anticipation of this disease have not existed previously because the virus is still relatively new. On March 2nd, 2020, Indonesia declared the first case of Indonesian citizens exposed to the COVID-19 virus. It has now been more than three years since COVID-19 has been in Indonesia. Medical experts from various countries have also been trying to create and develop ways to overcome this serious disease, one of which is the creation of vaccines for COVID-19. Generally, making a new vaccine actually takes a long time, which can take up to ten years [2].

Received 13 December 2023; Accepted 16 March 2024

On January 13th, 2021, COVID-19 vaccination has started in Indonesia with varying responses. Based on the survey that was conducted by the Ministry of Health, Indonesian Technical Advisory Group on Immunization (ITAGI), United Nations Children's Fund (UNICEF), and WHO before the COVID-19 vaccination distribution started, 45.7% of respondents (n=112,888) stated that they were not willing to get vaccinated [3]. It is because the respondents were not sure about the safety of the vaccine, afraid of side effects, religious beliefs, and other similar reasons.

Various responses that have appeared regarding the COVID-19 vaccination, both before and after the vaccination program, can be found through social media. One of the popularly used social media by Indonesians is Twitter. It is the fifth most popular social media application among Indonesians, with 63.6% of the population using it [4]. The total users indicate that it is possible for Indonesian citizen to express their reactions regarding the COVID-19 vaccination on Twitter. Therefore, sentiment analysis of COVID-19 vaccination can be conducted to help the government take further actions based on the results provided.

Naïve Bayes is a simple probability model that tends to work well on text classification tasks and usually takes less time to train compared to other models, such as Support Vector Machine [5]. Meanwhile, Multinomial Naïve Bayes is a variant of Naïve Bayes that is used for multinomially distributed data, as found in many text classification tasks [6]. In research conducted by Abbas, et al., for movie reviews using Multinomial Naïve Bayes, an accuracy of 90% was obtained with the help of the Term Frequency-Inverse Document Frequency (TF-IDF) method [7]. Then, another research with a similar task, namely sentiment analysis on COVID-19 vaccine news by using the Naïve Bayes algorithm was previously conducted with an accuracy of 73.75% [8].

Based on the above background, we aim to increase the performance of the Multinomial Naïve Bayes algorithm with the help of the TF-IDF method in analyzing sentiment towards COVID-19 vaccination in Indonesia. To support this implementation, the dataset that will be used is collected from tweets of Twitter users, starting from the first time COVID-19 vaccination was held in Indonesia.

2 Methods

2.1 Labelling

Labeling is the process of assigning labels or tags to raw data in order to signify the prediction that will be made by a Machine Learning algorithm. Labeling works by understanding the meaning of a sentence based on the context in which it is discussed, rather than by a word-by-word assessment [9]. There are several ways to do labeling, such as automated labeling and manual labeling. Automated labeling is a labeling that involves machines using the data that has been trained before, while manual labeling involves humans directly labeling the data, so it is the most effective way considering that humans are better at recognizing patterns in text data sets.

2.2 Sentiment Analysis

Sentiment analysis is the process of automatically understanding, extracting, and processing textual data and obtaining sentiment information contained in a sentence [10]. Normally it is started by conducting text pre-processing. Text pre-processing is an important stage in changing the form of unstructured text data into more structured text data [11]. It can be divided into seven steps [12], which are:

- Data Cleaning - a process of cleaning text data by removing regular expressions, such as punctuation marks, symbols, and numbers. This is done in order to reduce noise.
- Case Folding - a process of turning all characters in text data into lowercase.
- Normalization - a step to perform normalization on words that are not standard in Bahasa Indonesia. It is used to restore non-standard words or abbreviated words into standard words according to the rules of the Indonesian Dictionary.
- Stop word Removal - a step to remove words that are not important or do not affect the meaning of a sentence, for example, “which”, “to”, or “with”.
- Tokenization - a step to separate a sentence into smaller fragments which is per word.
- Stemming - a step to find the root forms of inflected words, but only by removing the suffixes or prefixes used in a word.
- Lemmatization - same as stemming, which is used to find the root forms of inflected words, but with a different method. It is actually not only removing the suffixes or prefixes but also returning root forms of stemmed words.

2.3 Term Frequency – Inverse Document Frequency (TF-IDF)

TF-IDF is a method to give weighted values to a term. It starts by calculating the number of times a term appears in a document (the term frequency or TF), and then calculating the number of documents that contain a term (the inverse document frequency or IDF) [12]. To calculate TF, IDF, and TF-IDF these equations (1)-(3) can be used [13].

$$tf(t,d) = \frac{n_{i,j}}{\sum_k n_{i,j}} \qquad (1)$$

$$idf(t) = \log\left(\frac{N}{df_t}\right) \qquad (2)$$

$$w_{t,d} = tf(t,d) \times idf(t) \qquad (3)$$

where $tf(t,d)$ is the term frequency, $n_{i,j}$ is the number of times a term appears in a document, $idf(t)$ is the inverse document frequency, N is the total number of documents, df_t is the number of documents that contain a specific term, and $w_{t,d}$ is the weighted term value in a document.

2.4 Multinomial Naïve Bayes

Multinomial Naïve Bayes classifier is an enhanced model of the Bayes algorithm that is suitable for text or document classification, especially when applied to large datasets [14]. It assumes that all attributes are independent of each other given the class context and ignores all dependencies between attributes. The following equation (4) can be used to perform the Multinomial Naïve Bayes classifier [11]:

$$P(c|term\ doc\ d) = P(c) \times P(t_1|c) \times P(t_2|c) \times \ldots \times P(t_n|c) \qquad (4)$$

To calculate $P(c)$ or also known as the prior probability we used equation (5) as follows:

$$P(c) = \frac{N_C}{N} \qquad (5)$$

Then, to calculate $P(t_i|c)$ or also known as the probability likelihood with Laplacian smoothing, we used equation (6):

$$P(t_i|c) = \frac{c(t_i,c)+K}{\sum_{t\in V} c(w,c)+|V|} \tag{6}$$

where V is the number of unique terms in the sample.

Whereas, to calculate probability likelihood while using TF- IDF, we used equation (7) as follows:

$$P(w_i|c) = \frac{W_{CW}+1}{(\sum W\prime \in V W\prime_{CW})+B\prime} \tag{7}$$

where W_{CW} is the TF-IDF weighted term value of term w in class c, and B' is the number of unique terms for which the IDF value is not multiplied with TF in all documents.

2.5 Confusion Matrix

Confusion matrix is used as a tool to evaluate the research that has been done. The evaluation metrics that will be calculated are accuracy, precision, recall, and F1-score. Accuracy is the ratio of the number of correctly classified cases in the test set divided by the total number of cases in the test set. Precision is the ratio of actual cases classified as positive to all conditions classified as positive. Meanwhile, recall is the ratio of actual cases classified as positive for all actual positive cases. F1-score is the average of precision and recall. Equations (8)-(11) show all respective metrics used in this study [15].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{8}$$

$$Precision = \frac{TP}{TP+FP} \tag{9}$$

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

$$F1-score = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{11}$$

2.6 k-fold Cross Validation

Cross-validation is a statistical method to estimate a Machine Learning predictive model performance [16]. One of the most commonly used techniques is k-fold cross-validation, which works by separating an existing dataset into a number of k partitions with the same number in each group so that the testing will also be carried out as much as the number of determined ‘k’ [17]. For real-world datasets, it is recommended to use stratified 10-fold cross-validation [18].

3 Research Methodology

In this section, a brief explanation of the research methodology is presented. Fig. 1 shows the main steps of the proposed method conducted in this research.

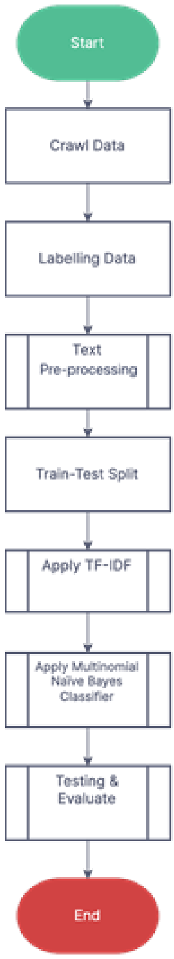

Fig. 1. Main flowchart

1) Data crawling

Data crawling was conducted by using the Twint library and searching with several keywords, namely “vaksin”, “vaksin covid”, and “#vaksin” starting from January 13th, 2021 to December 31st, 2021. From the crawling results, 4,418 tweets were obtained and then stored in .csv format to be processed in the labeling step.

2) Data labeling

The labeling method used in this research is manual labeling which was performed by three people as part of the Indonesian general public who receive information related to COVID-19 vaccination. These three people were given instructions as can be seen in Table 1. From the labeled tweets, each will be taken where the label is the most chosen one as the final label (majority voting). Suppose in tweet A, the first person chose “negative”, the second person chose “negative”, and the third person chose “neutral”, then the final label will be “negative”. Meanwhile, if all three people chose different labels, then the final label that will be taken is “neutral”. Of 4,418 tweets and the final label results, there are 1,844 positive tweets, 2,485 neutral tweets, and 88 negative tweets. The labeled dataset is then saved in .csv format.

Table 1: Labelling instructions

No	Instruction
1	Read the sentence in column ‘tweet’ and choose whether the sentence is positive, negative, or neutral

2	Make sure to type the label in lowercase and no typo
3	Positive: support, advice, constructive or encouraging, vaccine quotas, and others
4	Negative: distrust, rejection, slander, hoax, satire, complaints, do not want to receive vaccines, etc.
5	Neutral: ads, greetings, vaccine-related questions, to make decisions, stories outside the vaccine, not or in favor of vaccines

3) Text pre-processing

Text pre-processing was divided into two types. The first one is for Indonesian text and the second one is for the translated text, which was translated to English. It was translated because there are some tweets contain a mixture of languages, therefore language leveling was done. For Indonesian text, the steps are data cleaning, case folding, tokenization, stop word removal, stemming, and normalization. Stemming was performed with the Sastrawi library and normalization was performed by using a custom dictionary which contains all the words that are not up to standard and their standard form. Meanwhile, for English text, the steps are data cleaning, case folding, tokenization, stop word removal, and lemmatization. Lemmatization was performed with the help of the Spacy Library.

4) Apply TF-IDF

Before entering the TF-IDF step, the data will be split into training and testing data. The training data was used to calculate the TF, IDF, and TF-IDF based on equations (1) – (3). Meanwhile, the testing data was used for the testing phase later.

5) Apply Multinomial Naïve Bayes classifier

By using training data that has passed the term weighting model (Count vectorizer or TF-IDF), the value will be trained using the Multinomial Naïve Bayes classifier. Equations (5) – (7) were executed in order to build the model based on training data.

6) Test and evaluate

Based on the trained model, testing was conducted to predict the model's performance. To predict, equation (4) was used. The label with the highest value of posterior probability will become the predicted answer on the corresponding test data. Then, it will be evaluated and validated using a confusion matrix and stratified k-fold cross-validation.

4 Results and Discussion

In this section, an explanation of test scenarios that have been carried out is presented. All these scenarios test the use of TF-IDF and are compared with Count Vectorizer. The total data after going through the text pre-processing step is 4,141 tweets (2,255 neutral tweets, 1,799 positive tweets, and 87 negative tweets). The three different scenarios explored in this study are as follows:

1) Scenario 1: TF-IDF on original data – perform TF-IDF (and Count) vectorizer on original (unbalanced) data directly (4,418 tweets)
2) Scenario 2: TF-IDF on resampled data – perform TF-IDF (and Count) vectorizer on resampled (balanced) data (1,799 positive, 1,799 negative, and 1,799 neutral tweets)
3) Scenario 3: TF-IDF on translated data – perform TF-IDF (and Count) vectorizer on translated tweets data from Indonesian into English

4.1 TF-IDF on Original Data

The model performance results when performed on original data by using Count vectorizer and TF-IDF vectorizer (scenario 1) are shown in Tables 2 and 3. We used two hyperparameters available in scikitlearn, namely max_df and min_df, to determine the best model for different vectorizers applied in this study. max_df is used to remove terms that appear too frequently in the corpus, meanwhile, min_df is used to remove terms that appear too infrequently in the corpus.

Table 2: Test results for 80:20 or 5-fold on original data

Original Data (max_df = 0.9)				
Splitting	**Vector**	**min_df**	**Accuracy**	**F1-score**
Dataset 80:20	Count Vectorizer	1	69.84%	50.51%
		3	70.81%	51.29%
		5	69.60%	52.62%
	TF-IDF Vectorizer	1	70.93%	47.24%
		3	69.72%	46.72%
		5	68.40%	45.83%
5-fold CV	Count Vectorizer	1	70.71%	47.63%
		3	70.06%	50.00%
		5	69.55%	51.91%
	TF-IDF Vectorizer	1	70.51%	47.51%
		3	70.66%	47.35%
		5	69.77%	46.79%

Table 3: Test results for 90:10 or 10-fold on original data

Original Data (max_df = 0.9)				
Splitting	**Vector**	**min_df**	**Accuracy**	**F1-score**
Dataset 90:10	Count Vectorizer	1	69.16%	46.58%
		3	68.92%	46.43%
		5	69.68%	51.41%
	TF-IDF Vectorizer	1	71.08%	47.28%
		3	71.57%	47.96%
		5	69.16%	46.34%
10-fold CV	Count Vectorizer	1	70.85%	49.03%
		3	70.63%	51.32%
		5	69.81%	55.01%
	TF-IDF Vectorizer	1	71.19%	47.50%
		3	71.19%	47.72%
		5	70.42%	47.24%

Based on the results, it is found that the best accuracy and F1-score using TF-IDF are obtained when using a dataset with 90% training and 10% test ratio. In contrast to using Count Vectorizer, a dataset with 80% training and 20% test can give better accuracy and F1-score. However, while both are validated using stratified k-fold cross-validation, the accuracy and F1-score are better when using the value of k = 10. This is because the train-test split is only done once, so the model performance is lower due to the fact that it does not have the opportunity to take turns learning the unseen (not used) data.

Then, based on the min_df values that were tested, the highest accuracy value was obtained when using TF-IDF with min_df = 1. But, by looking at the F1-score value, the best result is obtained when using Count Vectorizer with min_df = 5. As can be observed from the results, the greater the min_df value of TF-IDF, then the smaller the F1-score is. Meanwhile, the greater min_df value of Count Vectorizer, then the F1-score is greater as well. It indicates that TF-IDF needs all words to describe the importance of the existing word so that if the word is removed, then it will result in lower accuracy and F1-score.

Because the data used for this research is imbalanced, the F1-score needs to be a metric that is considered. By looking at the confusion matrix of each label, TF-IDF indeed has more correct results in total than Count Vectorizer. Therefore, the accuracy of using TF-IDF is higher than the Count Vectorizer. However, the F1-score obtained by using TF-IDF seems to be dropping in value because TF-IDF could not predict correctly the negative tweet, so it impacted on lower F1-score compared to the Count Vectorizer.

4.2 TF-IDF on Resampled Data

Table 4 shows the results of testing on resampled data (scenario 2). Resampled data was obtained by over-sampling negative labeled data and down-sampling neutral labeled data to get a balanced dataset with a similar number of cases found in positive labeled data. Here we tested the values of k-fold = 10 and min_df = 1 and 5 based on the best min_df value and k-fold value found from the previous test.

Table 4: Test results for 90:10 or 10-fold on resampled data

Resampled Data (max_df = 0.9)				
Splitting	**Vector**	**min_df**	**Accuracy**	**F1-score**
Dataset 90:10	Count Vectorizer	1	82.59%	82.48%
		5	76.30%	76.12%
	TF-IDF Vectorizer	1	82.78%	82.54%
		5	77.41%	76.83%
10-fold CV	Count Vectorizer	1	83.05%	82.79%
		5	79.56%	79.18%
	TF-IDF Vectorizer	1	83.18%	82.81%
		5	79.97%	79.25%

Based on the results shown, it can be seen that the best accuracy and F1-score are obtained when using TF-IDF. Count Vectorizer gives high accuracy and F1-score as well, but not as high as TF-IDF. By looking at the min_df value, both TF-IDF and Count Vectorizer provide high results when using min_df = 1. Meanwhile, when using min_df = 5, there is a decrease in accuracy and F1-score. This is because a term that appears at least once in one document also has an influence on the classification, especially when using TF-IDF. TF-IDF gives a large weight to words that appear in one document and a small weight to words that appear in almost all documents. This is in contrast to Count Vectorizer which calculates word frequency by giving a weight of 1 to each occurrence in each document.

From the test conducted, it is known that using TF-IDF is better than using Count Vectorizer with min_df = 1 on the balanced dataset. The high results in this test are also due to resampling that was done before the data splitting process so that there would be some of the same data in the test data as in the training data and the same results are repeated as well.

4.3 TF-IDF on Translated Data

The results of this testing (scenario 3 using translated data from Indonesian to English) can be seen in Table 5. Based on the results, it has the same results as the first scenario on the original data, where the highest accuracy is obtained when using TF-IDF with a value of 70.03% and min_df = 1. Meanwhile, looking at the F1-score, the best result is obtained when using Count Vectorizer with a value of 52.09% and min_df = 5. However, after comparing the results to the testing on original data, it was found that the classification is better when using its original language, both in terms of accuracy and F1-score. This could happen due to differences in the text pre-processing between the Indonesian and English languages.

Table 5: Test results for 90:10 or 10-fold on translated data

Resampled Data (max_df = 0.9)				
Splitting	**Vector**	**min_df**	**Accuracy**	**F1-score**
Dataset 90:10	Count Vectorizer	1	68.19%	45.83%
		5	67.47%	51.24%
	TF-IDF Vectorizer	1	68.19%	45.21%
		5	68.19%	45.55%
10-fold CV	Count Vectorizer	1	69.57%	48.14%
		5	68.61%	52.09%
	TF-IDF Vectorizer	1	70.03%	46.68%
		5	68.90%	46.17%

4.4 Limitations and Future Directions

There are some limitations in this study. First, we used a manual labeling process involving three different persons due to the limited time frame and resources to use automatic labeling. Future research could utilize automatic labeling methods (for Indonesian) and compare the results found by using the manual labeling process. Next, as can be seen from Scenario 2, it is recommended to use a balanced dataset with varied words in order to improve the performance of TF-IDF. The application of the n-gram method [19] to increase the variation of pre-processed words can be considered. It is also recommended to try using other classifier models, such as BERT (Bidirectional Encoder Representations from Transformers) and Long Short-Term Memory (LSTM) models [20] which can distinguish the meaning of words to improve the classification performance. Lastly, in this study, our focus is on the model development using Multinomial Naïve Bayes and different vectorizers to investigate the COVID-19 vaccination sentiment in Indonesia. Another research direction to create a decision-making tool (DMT) [21] for practical usage of the built model can also be done in the future.

5 Conclusion

Based on the experimental results that have been conducted to manually labeled data with a total of 4,141 tweets after going through text pre-processing, the implementation of Multinomial Naïve Bayes with Count Vectorizer could classify better than TF-IDF if considering the label results of correct classifications. Meanwhile, if considering the total number of correct classifications, then TF-IDF is better with an accuracy of 71.19% and min_df = 1. For the implementation of Multinomial Naïve Bayes on a balanced dataset (5,397 tweets after resampling), the best classification is obtained when using TF-IDF with min_df = 1, which in terms of accuracy and F1-score reached 83.18% and 82.81%, respectively.

ACKNOWLEDGEMENTS
The authors would like to acknowledge the support given by Universitas Multimedia Nusantara during this study.

References

[1] S. Hansun, V. Charles, and T. Gherman, "The Role of the Mass Vaccination Programme in Combating the COVID-19 Pandemic: An LSTM-based Analysis of COVID-19 Confirmed Cases," *Heliyon*, vol. 9, no. 3, p. e14397, Mar. 2023, doi: 10.1016/j.heliyon.2023.e14397.

[2] I. P. Sari and S. Sriwidodo, "Perkembangan Teknologi Terkini dalam Mempercepat Produksi Vaksin COVID-19," *Maj. Farmasetika*, vol. 5, no. 5, p. 204, Aug. 2020, doi: 10.24198/mfarmasetika.v5i5.28082.

[3] The Ministry of Health, NITAG, UNICEF, and WHO, "COVID-19 Vaccine Acceptance Survey in Indonesia," Indonesia, 2020. [Online]. Available: https://covid19.go.id/storage/app/media/Hasil Kajian/2020/November/vaccine-acceptance-survey-en-12-11-2020final.pdf.

[4] S. Kemp, "Digital 2021: Indonesia," 2021. https://datareportal.com/reports/digital-2021-indonesia.

[5] V. Narayanan, I. Arora, and A. Bhatia, "Fast and Accurate Sentiment Classification Using an Enhanced Naive Bayes Model," 2013, pp. 194–201.

[6] S. Xu, Y. Li, and Z. Wang, "Bayesian Multinomial Naïve Bayes Classifier to Text Classification," 2017, pp. 347–352.

[7] M. Abbas, K. A. Memon, A. A. Jamali, S. Memon, and A. Ahmed, "Multinomial Naive Bayes Classification Model for Sentiment Analysis," *Int. J. Comput. Sci. Netw. Secur.*, vol. 19, no. 3, pp. 62–67, 2019, [Online]. Available: http://paper.ijcsns.org/07_book/201903/20190310.pdf.

[8] A. Suwarno and A. Andriani, "Analisis Sentimen pada Media Sosial Twitter Mengenai Tanggapan Vaksinasi COVID-19 Menggunakan Metode Naive Bayes," *J. Tek. Ind.*, vol. 2, no. 2, pp. 22–29, 2021, [Online]. Available: https://jurnal.pelitabangsa.ac.id/index.php/JUTIN/article/view/906.

[9] N. Anggraini, E. S. N. Harahap, and T. B. Kurniawan, "Text Mining - Text Analysis Related to COVID-19 Vaccination Issues," *J. Iptek-Kom*, vol. 23, no. 2, pp. 141–153, 2021, [Online]. Available: https://jurnal.kominfo.go.id/index.php/iptekkom/article/view/4259.

[10] G. A. Buntoro, "Sentiment Analysis to Prediction DKI Jakarta Governor 2017 on Indonesian Twitter," *Int. J. Sci. Eng. Inf. Technol.*, vol. 2, no. 2, pp. 36–40, 2018, [Online]. Available: https://journal.trunojoyo.ac.id/ijseit/article/view/2744.

[11] A. Rahman, W. Wiranto, and A. Doewes, "Online News Classification Using Multinomial Naive Bayes," *ITSMART J. Teknol. dan Inf.*, vol. 6, no. 1, pp. 32–38, 2017, [Online]. Available: https://jurnal.uns.ac.id/itsmart/article/view/11310.

[12] W. A. Luqyana, I. Cholissodin, and R. S. Perdana, "Analisis Sentimen Cyberbullying pada Komentar Instagram dengan Metode Klasifikasi Support

Vector Machine,” *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, vol. 2, no. 11, pp. 4704–4713, 2018, [Online]. Available: https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/3051.

[13] H. Zhou, “Research of Text Classification Based on TF-IDF and CNN-LSTM,” *J. Phys. Conf. Ser.*, vol. 2171, no. 1, p. 012021, Jan. 2022, doi: 10.1088/1742-6596/2171/1/012021.

[14] A. H. Setianingrum, D. H. Kalokasari, and I. M. Shofi, “Implementasi Algoritma Multinomial Naive Bayes Classifier,” *J. Tek. Inform.*, vol. 10, no. 2, pp. 109–118, Jan. 2018, doi: 10.15408/jti.v10i2.6822.

[15] D. Lovell, D. Miller, J. Capra, and A. P. Bradley, “Never Mind the Metrics-What about the Uncertainty? Visualising Binary Confusion Matrix Metric Distributions to Put Performance in Perspective,” in *Proceedings of the 40th International Conference on Machine Learning*, 2023, p. PMLR 202:22702-22757, [Online]. Available: https://proceedings.mlr.press/v202/lovell23a.html.

[16] F. Tempola, M. Muhammad, and A. Khairan, “Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 5, pp. 577–584, Oct. 2018, doi: 10.25126/jtiik.201855983.

[17] R. Rismanto, D. W. Wibowo, and A. R. Syulistyo, “Implementation of Naive Bayes Classifier and Log Probabilistic for Book Classification Based on the Title,” *Int. J. Eng. Technol.*, vol. 7, no. 4.36, pp. 1361–1366, 2018, [Online]. Available: https://www.sciencepubco.com/index.php/ijet/article/view/28987.

[18] D. Berrar, “Cross-Validation,” in *Encyclopedia of Bioinformatics and Computational Biology*, Elsevier, 2019, pp. 542–545.

[19] Y. Vernanda, S. Hansun, and M. B. Kristanda, “Indonesian Language Email Spam Detection using N-gram and Naïve Bayes Algorithm,” *Bull. Electr. Eng. Informatics*, vol. 9, no. 5, pp. 2012–2019, Oct. 2020, doi: 10.11591/eei.v9i5.2444.

[20] S. Hansun, A. Suryadibrata, R. Nurhasanah, and J. Fitra, “Tweets Sentiment on PPKM Policy as a COVID-19 Response in Indonesia,” *Indian J. Comput. Sci. Eng.*, vol. 13, no. 1, pp. 51–58, Feb. 2022, doi: 10.21817/indjcse/2022/v13i1/221301302.

[21] S. Hansen, A. Suryadibrata, and S. Hansun, “Infrastructure Project Selection Automation using Non-structural Fuzzy Decision Support System II,” *Eastern-European J. Enterp. Technol.*, vol. 1, no. 3 (121), pp. 46–56, Feb. 2023, doi: 10.15587/1729-4061.2023.271822.

Notes on contributors

Liana, a dedicated Data Analyst, graduated with a Bachelor’s degree in Informatics from Multimedia Nusantara University. With over 2 years of experience in the field, Liana is skilled in SQL, Python, and various data visualization tools such as Metabase and AWS Quicksight. She has a strong problem-solving and analytical skills. Currently, she works at a FinTech company, contributing to delivering insights and analysis based on data.

Seng Hansun received the Bc. degree in Mathematics (S.Si.) from Universitas Gadjah Mada, Yogyakarta, in 2008 and Master of Computer Science (M.Cs.) degree from the same university in 2011. Since then, he has been a Lecturer with the Computer Science Department, Universitas Multimedia Nusantara (UMN), Indonesia. He had been appointed as the Information and Communication Technology (ICT) Faculty Research Coordinator, Deputy Head of Computer Science Department, and Head of Informatics Department at UMN. He had published two books and more than 140 articles during his career as both academician and researcher at UMN. His research interests lately include computational science, soft computing methods, and internet and mobile technology in various fields, especially in the Medical Informatics area.