A Hybrid CNN-LSTM Model With Word-Emoji Embedding For Improving The Twitter Sentiment Analysis on Indonesia's PPKM Policy

Syafrial Fachri Pane
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
fachrie@student.telkomuniversity.ac.id

Jenly Ramdan
Bachelor of Applied Informatics Engineering
Universitas Logistik dan Bisnis International
Bandung, Indonesia
jenly.ramdan@poltekpos.ac.id

Aji Gautama Putrada
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
ajigps@telkomuniversity.ac.id

Mohamad Nurkamal Fauzan
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
mnurkamalfauzan@student.telkomuniversity.ac.id

Rolly Maulana Awangga
Bachelor of Applied Informatics Engineering
Universitas Logistik dan Bisnis International
Bandung, Indonesia
awangga@ulbi.ac.id

Nur Alamsyah
Advanced and Creative Networks Research Center
Telkom University
Bandung, Indonesia
nuralamsyah@student.telkomuniversity.ac.id

***Abstract*—The policy of limiting community mobilization is implemented to reduce the daily rate of COVID-19. However, a high-accuracy sentiment analysis model can determine public sentiment toward such policies. Our research aims to improve the accuracy of the LSTM model on sentiment analysis of the Jakarta community towards PPKM using Indonesian language Tweets with emoji embedding. The first stage is modeling using the hybrid CNN-LSTM model. It is a combination between CNN and LSTM. The CNN model cites word embedding and emoji embedding features that reflect the dependence on temporary short-term sentiment. At the same time, LSTM builds long-term sentiment relationships between words and emojis. Next, the model evaluation uses *Accuracy*, *Loss*, the receiver operating curve (ROC), the precision and recall curve, and the area under curve (*AUC*) value to see the performance of the designed model. Based on the results of the tests, we conclude that the CNN-LSTM Hybrid Model performs better with the words+emoji dataset. The ROC AUC is 0.966, while the precision-recall curve AUC is 0.957.**

***Index Terms*—Sentiment Analysis, Tweet, Machine Learning, Model Hybrid, CNN-LSTM**

I. Introduction

India, Hong Kong, Shanghai, the UK, Brussels, and Australia are some countries affected after implementing the COVID-19 rate control, namely the lockdown. The aftermath of the impact of the lockdown in various parts of the country was the emergence of a protest movement against the lockdown policy with the anti-lockdown hashtag on Twitter and social media. Therefore, a sentiment analysis model with high accuracy is needed to understand the direction of public opinion on the implemented COVID-19 rate control policy. So that later it can be anticipated if there is a policy rejection in the community and as a government benchmark for consideration in determining policies to control the rate of COVID-19, namely physical distancing or Community Activities Restrictions Enforcement (PPKM). PPKM in Indonesia is the Government's policy to deal with the COVID-19 pandemic in Indonesia.

In previous studies, sentiment analysis research related to PPKM policies has been carried out using a twitter dataset which consists of a variable, namely tweet. Pane *et al.* used long short-term memory (LSTM) to classify tweets into negative and positive sentiments and achieved an accuracy score of 0.917 [1]. On the other hand, some research use emoji embedding to improve tweet sentiment classification performance [2]. The emoji embedding is used to change the emoji into the form of numbers which will then be converted into emoji embedding [3]. Later this emoji embedding will be used as input for modeling which becomes a vector representation of words or emojis that have semantically similar meanings [4]. An improvement using emoji embedding which results in an increase in model accuracy from the previous model is a research opportunity.

Our research proposes using emoji embedding to increase the Twitter sentiment analysis performance on PPKM policies with a hybrid CNN-LSTM model. The first stage is modeling using the hybrid CNN-LSTM model. It is a combination between CNN and LSTM. The CNN model cites word embedding and emoji embedding features that reflect the dependence on temporary short-term senti-

979-8-3503-9961-5/22/$31.00 ©2022 IEEE

ment. At the same time, LSTM is used to build long-term sentiment relationships between words and emojis. Next, the model evaluation uses *Accuracy*, *Loss*, the receiver operating curve (ROC), the precision and recall curve, and the area under curve (AUC) value to see the performance of the designed model.

To the best of our knowledge, no other research has conducted a hybrid CNN-LSTM model with emoji embedding on twitter sentiment analysis from Indonesia's PPKM policy. The contributions of our research are as follows:

1) A proven words+emoji embedding method performs better than the legacy words dataset
2) A method that deals with data imbalances that occur in datasets using random oversampling, ROC, and precision-recall curves
3) A good hybrid CNN+LSTM model using the softmax activation function and SGD optimization

The contents of this paper have the following systematics: Section II discusses related work. Section III describes the Research Methodology. Section IV reports the test results and discusses them with previous studies. Finally, Section V highlights the important results of this study.

II. Related Work

A study has discussed improving the accuracy of the LSTM model using word-emoji embedding [5] by utilizing emoji contained in tweets to overcome semantic limitations. So that the tweet emojis translate into a description which then combines with the text in the tweet. The result is a combined word representation of the text and emoji in the tweet. Adding emoji preprocessing to the embedding layer for the input model where this emoji preprocessing enriches the semantics so that the model f1, accuracy, precision, and recall, increases by 0.933, 0.959, 1.0, and 0.923, respectively. Li *et al.* [6] used emoji embedding on Weibo texts for sentiment analysis with a Bi-directional gated recurrent unit (GRU). The study case was on Chinese communities. The result of the study showed that emoji-attention GRU (EAGRU) performed better than GRU. The best performance was an f1-score of 0.81 on negative sentiments.

Furthermore, Henessa *et al.* [7] used secondary datasets of Indonesia's PPKM Tweets and compared several classification methods for sentiment analysis. The research used the term frequency-inverse document frequency (TF-IDF) for feature extraction. Logistic regression had the best performance compared to other methods, with an accuracy value of 87.5. Furthermore, Wahyuni *et al.* [8] used the Fuzzy Sugeno method to apply sentiment analysis on PPKM Policy tweet reactions. With, K-fold cross-validation, the research reached an accuracy of 0.89. Using emoji embedding on Indonesia's PPKM policy tweet is a research opportunity.

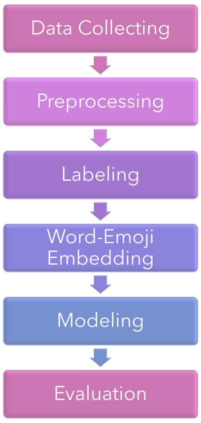

Fig. 1. The proposed research methodology.

III. Methodology

Fig. 1 explains the flow of our research. This section discusses the steps taken by the author in the research, which includes several steps starting from the data collection, preprocessing, labeling, word-emoji embedding, modeling, and evaluation.

A. Preprocessing and Emoji Embedding

Here we use a dataset of 3000 tweets collected from September – November 2021 with the DKI Jakarta geo code as in Table I.

Fig. 2 explain the flow of preprocessing. This section discusses the steps taken by us in the preprocessing stage. Preprocessing is a process where datasets are collected from the media social twitter will be cleaned of unneeded elements. so that the data will be clean, quality and as desired. The preprocessing process is divided into several stages, that is translating, case folding, emoji process, remove punctuation, tokenization, remove stopwords and stemming [9].

The first two steps are translating and case folding. Some research shows that translating Indonesian tweets improve the sentiment analysis performance [10]. One preprocessing step is translating the Indonesian language dataset into English. Case folding uniforms the case in a sentence [11]. Such as punctuation and Unicodes. This process consists of four stages: cleaning punctuation, clearing numbers, changing all uppercase letters to lowercase letters, and cleaning excess spaces as in Table II.

The emoji process consists of three stages: deleting text from the dataset, describing emoji into descriptive text, and merging text and emoji descriptions. Table III

TABLE I
Dataset Specifications

Parameter	Value
Dataset Size	3000
Language	Indonesian Language
Date Range	September - November 2021
Source	Jakarta geo code

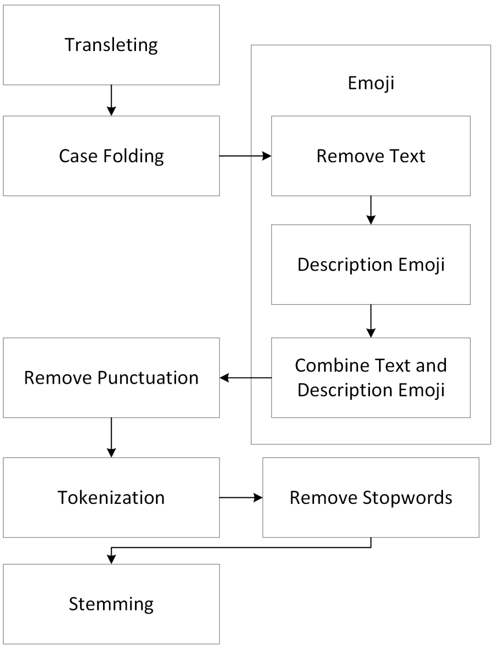

Fig. 2. Preprocessing Flow.

shows the example of each case. The deleting text process removes text and leaves the emoji. The describing emoji process inserts emoji description into an emoji character. Lastly, the merging text and emoji description stage replaces the emoji in the original text with the emoji description.

The three next processes are tokenization, removing stopwords, and stemming. Tokenization is a process of separating words, symbols, phrases, and entities from a text [12]. In this process, we use the help of the natural language toolkit (NLTK) library from Python. The remove stopwords stage is a removing conjunction process from a sentence such as which, and, and but [13]. In this stopword deletion process, first, it is done to define the will-be deleted words when this process executes—the NLTK library assists in removing this stopword. The stemming stage is removing affixes from words in the dataset [14]—the NLTK library assists in stemming. Table IV shows the results of tokenizing, removing stopwords, and stemming.

TABLE II
TRANSLATING AND CASE FOLDING STEPS

No	Process	Example Sentence Before Process	Example Sentence After Process
1	Translate	Saat PON Papua Covid menurun,	When PON Papua Covid is declining,
2	Cleaning punctuation	When PON Papua Covid is declining,	When PON Papua Covid is declining
3	Clearing numbers	because I can’t watch #PPKM Vol 1	because I can’t watch #PPKM Vol
4	Lower-casing	When PON Papua Covid is declining,	when pon papua covid is declining

TABLE III
THE EMOJI EMBEDDING PROCESSES

No	Process	Example Sentence Before Process	Example Sentence After Process
1	Deleting text	it will definitely decrease it means fix covid likes sports 🤣🤣	🤣🤣
2	Describing Emoji	🤣🤣	rolling on the floor laughing rolling on the floor laughing
3	Merging text and emoji descriptions	it will definitely decrease it means fix covid likes sports 🤣🤣	it will definitely decrease it means fix covid likes sports rolling on the floor laughing rolling on the floor laughing

After preprocessing, the next process labels each tweet with a positive or negative label based on polarity and subjectivity using TextBlob library [15]. Sentiment polarity describes the orientation of a tweet, for example, whether the tweet is negative, positive, or neutral [16]. Usually, there are three categories for sentiment polarity, but here we only use negative and positive polarity [17]. The equation is as follows:

$$Sentiment = \begin{cases} positive, & \text{if } Polarity \geq 0 \\ negative, & \text{if otherwise} \end{cases} \tag{1}$$

Furthermore, sentiment subjectivity explains quantitatively what proportion of a tweet is an opinion and what proportion is factual [18]. A high subjectivity value means that the tweet contains more personal opinions than

TABLE IV
THE TOKENIZATION, REMOVE STOPWORDS, AND STEMMING PROCESS

No	Process	Example Sentence Before Process	Example Sentence After Process
1	Tokenization	just eating and sleeping i hope its finished soon thumbs up light skin tone	[’just’, ’eating’, ’and’, ’sleeping’, ’i’, ’hope’, ’its’, ’finished’, ’soon’, ’thumbs’, ’up’, ’light’, ’skin’, ’tone’]
2	Remove stopwords	[’just’, ’eating’, ’and’, ’sleeping’, ’i’, ’hope’, ’its’, ’finished’, ’soon’, ’thumbs’, ’up’, ’light’, ’skin’, ’tone’]	[’eating’, ’sleeping’, ’hope’, ’finished’, ’soon’, ’thumbs’, ’light’, ’skin’, ’tone’]
2	Stemming	[’eating’, ’sleeping’, ’hope’, ’finished’, ’soon’, ’thumbs’, ’light’, ’skin’, ’tone’]	[’eat’, ’sleep’, ’hope’, ’finish’, ’soon’, ’thumb’, ’light’, ’skin’, ’tone’]

TABLE V
Sentiment Labeling Examples with Polarity and Subjectivity

No	Tweet	Polarity	Subjectivity	Sentiment
1	the government wants the economy to continue to move	0.1	0.05	positive
2	cc victim of disaster help omnibuslawtetapberku	-0.075	0.05	negative
3	good morning sister all how brother s brother s	0.35	0.35	positive
4	this ppkm fuck.	-0.4	0.6	negative

factual information [19]. Table V shows example results of the sentiment labeling. There are four examples. Each represents a possible sentiment of a tweet: low subjectivity - positive sentiment, low subjectivity - negative sentiment, high subjectivity - positive sentiment, and high subjectivity - negative sentiment.

Furthermore, we split the dataset into 50% training and 50% testing data. In the training data, 1053 have a positive sentiment label, while 53 have negative sentiments. The imbalance ratio (IR) is 4.79%, which has a moderate imbalance data degree [20]. To overcome the imbalance, we use the random oversampling method. The random oversampling method creates new random data samples to the minority label until the dataset is balanced. After the process, there are 1053 data for each positive and negative sentiment.

B. Hybrid CNN-LSTM Model

In this study, we propose an emoji-embedding CNN-LSTM for sentiment analysis. The CNN-LSTM model consists of an initial Convolutional layer which will receive word embeddings for each token in the title as input. The output generated from CNN layers will be aggregated to a smaller dimension and then fed into the LSTM layer, which extracts local features. Fig. 3 is an overview of our proposed CNN-LSTM architecture.

After going through the preprocessing stage, the words and emojis that are semantically related will be changed into numbers and made into an embedding layer. Later this embedding layer will be used as input for the LSTM sentiment classification method and then converted into vectors to find semantic similarities.

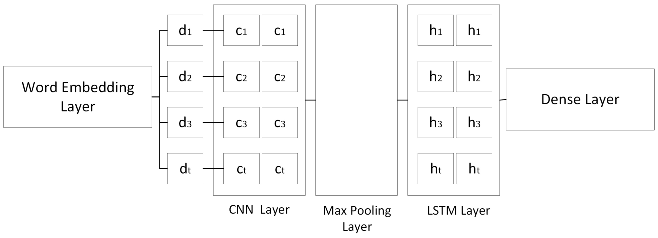

Fig. 3. Model Architecture.

The following is the formula for the proposed CNN-LSTM model:

$$\mathrm{z}^{1} = \mathrm{h}^{1-1} * \ \mathrm{W}^{1} \tag{2}$$

$$h_{xy}^{1} = \max_{\mathrm{i=0..s_j=0..s}} \mathrm{h}^{1-1}(\mathrm{x+i})(\mathrm{y+j}) \tag{3}$$

$$ft = \sigma(Wf \times [xt + ht - 1] + bf) \tag{4}$$

$$it = \sigma(Wi \times [xt + ht - 1] + bi) \tag{5}$$

$$C^{\sim}t = \tanh(WC \times [xt + ht - 1] + bC) \tag{6}$$

For the output layer, we use the softmax activation function. Softmax is a generalization of the logarithmic function to multiple dimensions [21]. In other words, softmax performs a standard exponential operation on its input and normalizes each class so that the sum of the results of all classes is equal to 1. The formula for softmax activation is as follows:

$$\sigma(x)_i = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}, i \in K \tag{7}$$

where $\sigma(x)_i$ is the softmax activation function for input x, K is the number of classes, e^x is the standard exponential function for input x.

The deep learning techniques have many optimization methods in the learning methods. Here we use stochastic gradient descent (SGD). SGD replaces the gradient descent calculation of the overall loss function with a stochastic value [22]. The formula for SGD is as follows:

$$w := w - \eta \nabla Q_i(w) \tag{8}$$

where w is the weight, η is the learning rate, $:=$ is the assignment operation, ∇ is the vector differential operator, and $Q_i(w)$ is the neuron function for the data item i.

C. Evaluation

We use *Accuracy* and *Loss* for the training process for evaluation. *Accuracy* is the composition of all the truly predicted data to all the data. The truly predicted data consists of true positives (TP) and true negatives (TN). Then *Loss* has the following formula:

$$loss = -(t_p log(f(s)_p) + t_n log(f(s)_n)) \tag{9}$$

where p is the positive sentiment class, n is the negative sentiment class, t is the actual value, $f(s)$ is the predicted result.

We want to compare the emoji embedding model with the generic model in the sentiment analysis case. Because the data is imbalanced, we use ROC, the precision-recall

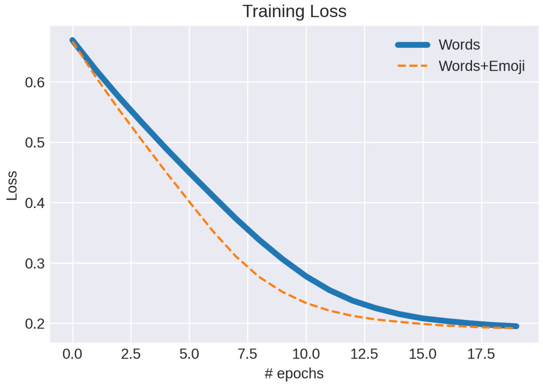

Fig. 4. The words+emoji model training loss compared to the words model.

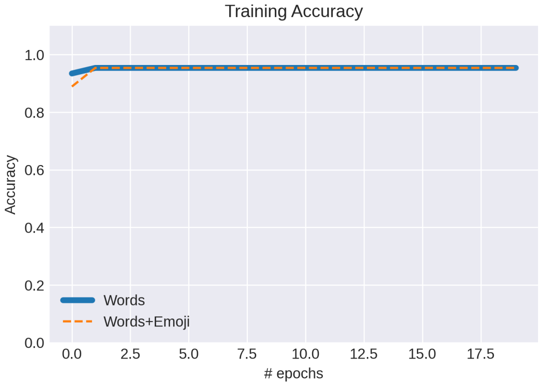

Fig. 5. The words+emoji model training accuracy compared to the words model.

curve, and the AUC to compare the performances. The ROC explains the correlation between the true positive rate (TPR) and the false positive rate (FPR) when dealing with the predicted probability results of the testing input data [23].

Furthermore, the precision-recall curve is similar to the ROC but uses precision and recall in the place of TPR and FPR. Then the quantitative performance of the two curves is the AUC, which is the area under each curve [24]. The AUC equation is as follows:

$$AUC = \sum_{i=1}^{N} \frac{x_i + x_{i-1}}{2} \times (y_i - y_{i-1}) \tag{10}$$

where N is the number of predicted probability results, x is the metric of the x-axis, and y is the metric of the y-axis.

IV. Results and Discussion

A. Results

We evaluate the fit model with the Keras library with the *Loss* curve and *Accuracy* curve. Based on the evaluation of our tried model fit, we find that the CNN LSTM model of the word+emoji dataset has a lower *Loss* curve and an approximate *Accuracy* curve than the word dataset. Fig 4 and Fig 5 are the results of model fit evaluation based on the *Loss* curve value and *Accuracy* curve value, respectively.

Based on the image in Fig. 6 and Fig. 7, the evaluation of model fit shows that the CNN-LSTM model with words+emoji dataset results ROC and precision-recall curve is higher than the model with the words dataset. After evaluating the fit model, we calculate the AUC. The goal is to find out how big the AUC of the two models are. From this study we find that the CNN-LSTM model with word+emoji dataset has superior AUC compared to the word dataset. The words+emoji dataset has an ROC $AUC = 0.966$, whereas the words dataset has an ROC AUC of 0.946. At the same time, the words+emoji dataset has a precision-recall curve $AUC = 0.957$, whereas the words dataset has a precision-recall curve AUC of 0.932.

B. Discussion

Beberapa penelitian telah menelaah penggunaan hybrid CNN-LSTM untuk sentiment analysis, seperti paper [25]. Model tersebut merupakan model dengan kinerja terbaik dibandingkan model lain nya, yaitu $Accuracy = 0.83$. Pane *et al.* menggunakan model yang sama untukanalyze public sentiment towards PPKM policies using Twitter data dan mendapatkan akurasi lebih tinggi, yaitu 0.94. Our research enhances the method of that paper with several methods. Compared to these studies, our study makes a three-fold contribution. First, we use words+emoji embedding, which performs better than the legacy words dataset. Second, we deal with imbalanced data in the dataset by using random oversampling, ROC, and precision-recall curves. Third, we use the softmax activation function and SGD optimization for a better CNN-LSTM hybrid model.

V. Conclusion

We created a hybrid CNN-LSTM model to predict public sentiment towards the PPKM policy issued by the government to deal with COVID-19. Our hybrid model uses the words+emoji dataset and uses emoji embedding in the process. Based on the results of the tests, we

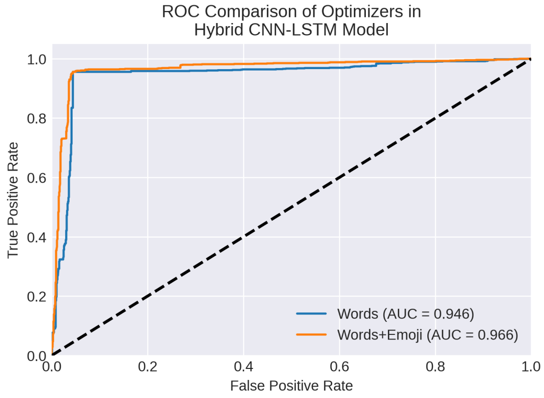

Fig. 6. The words+emoji model ROC compared to the words model.

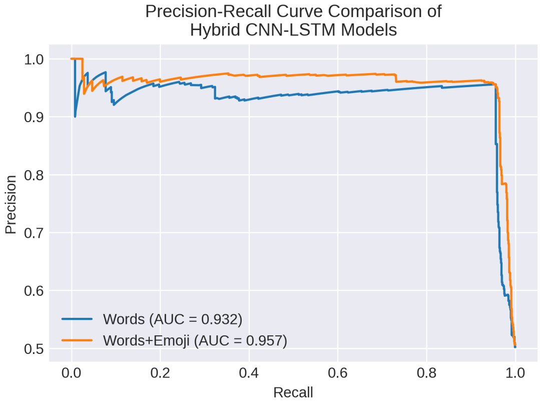

Fig. 7. The words+emoji model precision-recall curve compared to the words model.

conclude that the CNN-LSTM Hybrid Model performs better with the words+emoji dataset. The ROC AUC is 0.966, while the precision-recall curve AUC is 0.957.

References

[1] S. F. Pane and J. Ramdan, "Pemodelan machine learning: Analisis sentimen masyarakat terhadap kebijakan ppkm menggunakan data twitter," *Jurnal Sistem Cerdas*, vol. 5, no. 1, pp. 12–20, 2022.

[2] Y. Chen, J. Yuan, Q. You, and J. Luo, "Twitter sentiment analysis via bi-sense emoji embedding and attention-based lstm," in *Proceedings of the 26th ACM international conference on Multimedia*, pp. 117–125, 2018.

[3] M. Shardlow, L. Gerber, and R. Nawaz, "One emoji, many meanings: A corpus for the prediction and disambiguation of emoji sense," *Expert Systems with Applications*, vol. 198, p. 116862, 2022.

[4] A. Kumar, S. R. Sangwan, A. K. Singh, and G. Wadhwa, "Hybrid deep learning model for sarcasm detection in indian indigenous language using word-emoji embeddings," *Transactions on Asian and Low-Resource Language Information Processing*, 2022.

[5] C. Liu, F. Fang, X. Lin, T. Cai, X. Tan, J. Liu, and X. Lu, "Improving sentiment analysis accuracy with emoji embedding," *Journal of Safety Science and Resilience*, vol. 2, no. 4, pp. 246–252, 2021.

[6] D. Li, R. Rzepka, M. Ptaszynski, and K. Araki, "Emoji-aware attention-based bi-directional gru network model for chinese sentiment analysis.," in *LaCATODA/BtG@ IJCAI*, pp. 11–18, 2019.

[7] R. P. Henessa, M. A.-F. Fisabilillah, and W. R. Azizah, "Detection of public sentiment analysis model on the implementation of ppkm in indonesia," in *Proceedings of The International Conference on Data Science and Official Statistics*, vol. 2021, pp. 289–295, 2021.

[8] D. Wahyuni, E. Sumarminingsih, and S. Astutik, "Covid-19 vaccination and ppkm policy with the implementation of the fuzzy sugeno method to income classification," *JTAM (Jurnal Teori dan Aplikasi Matematika)*, vol. 6, no. 4, pp. 937–946, 2022.

[9] A. G. Putrada, I. D. Wijaya, and D. Oktaria, "Overcoming data imbalance problems in sexual harassment classification with smote," *International Journal on Information and Communication Technology (IJoICT)*, vol. 8, no. 1, pp. 20–29, 2022.

[10] Y. Yunitasari, A. Musdholifah, and A. K. Sari, "Sarcasm detection for sentiment analysis in indonesian tweets," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 13, no. 1, pp. 53–62, 2019.

[11] S. Bauskar, V. Badole, P. Jain, and M. Chawla, "Natural language processing based hybrid model for detecting fake news using content-based features and social features," *International Journal of Information Engineering and Electronic Business*, vol. 11, no. 4, pp. 1–10, 2019.

[12] A. Rai and S. Borah, "Study of various methods for tokenization," in *Applications of Internet of Things*, pp. 193–200, Springer, 2021.

[13] S. Behera, "Implementation of a finite state automaton to recognize and remove stop words in english text on its retrieval," in *2018 2nd international conference on trends in electronics and informatics (ICOEI)*, pp. 476–480, IEEE, 2018.

[14] D. Soyusiawaty, A. H. S. Jones, and N. L. Lestariw, "The stemming application on affixed javanese words by using nazief and adriani algorithm," in *IOP Conference Series: Materials Science and Engineering*, vol. 771, p. 012026, IOP Publishing, 2020.

[15] F. Illia, M. P. Eugenia, and S. A. Rutba, "Sentiment analysis on pedulilindungi application using textblob and vader library," in *Proceedings of The International Conference on Data Science and Official Statistics*, vol. 2021, pp. 278–288, 2021.

[16] A. Kumar and D. Gupta, "Sentiment analysis as a restricted nlp problem," in *Natural Language Processing for Global and Local Business*, pp. 65–96, IGI Global, 2021.

[17] S. Kausar, X. Huahu, W. Ahmad, and M. Y. Shabir, "A sentiment polarity categorization technique for online product reviews," *IEEE Access*, vol. 8, pp. 3594–3605, 2019.

[18] A. S. Neogi, K. A. Garg, R. K. Mishra, and Y. K. Dwivedi, "Sentiment analysis and classification of indian farmers' protest using twitter data," *International Journal of Information Management Data Insights*, vol. 1, no. 2, p. 100019, 2021.

[19] Z. Zhao, Z. Zhang, and F. Hopfgartner, "Ss-bert: Mitigating identity terms bias in toxic comment classification by utilising the notion of" subjectivity" and" identity terms"," *arXiv preprint arXiv:2109.02691*, 2021.

[20] P. Vuttipittayamongkol, E. Elyan, and A. Petrovski, "On the class overlap problem in imbalanced data classification," *Knowledge-based systems*, vol. 212, p. 106631, 2021.

[21] F. De Pretis and J. Landes, "Ea3: A softmax algorithm for evidence appraisal aggregation," *PLoS One*, vol. 16, no. 6, p. e0253057, 2021.

[22] A. Chaudhuri, "The minimization of empirical risk through stochastic gradient descent with momentum algorithms," in *Computer Science On-line Conference*, pp. 168–181, Springer, 2019.

[23] A. G. Putrada and M. Abdurohman, "Anomaly detection on an iot-based vaccine storage refrigerator temperature monitoring system," in *2021 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA)*, pp. 75–80, IEEE, 2021.

[24] A. G. Putrada and D. Perdana, "Improving thermal camera performance in fever detection during covid-19 protocol with random forest classification," in *2021 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*, pp. 1–6, IEEE, 2021.

[25] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, "Co-lstm: Convolutional lstm model for sentiment analysis in social big data," *Information Processing & Management*, vol. 58, no. 1, p. 102435, 2021.