Exploring the Performance of BERT Models for Multi-Label Hate Speech Detection on Indonesian Twitter

Muhammad Razi Mahardika
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
muhammad.mahardika004@binus.ac.id

I Putu Janardana Wijaya
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
i.wijaya005@binus.ac.id

Arvin Rayhandi Prayoga
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
arvin.prayoga@binus.ac.id

Henry Lucky
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
henry.lucky@binus.ac.id

Irene Anindaputri Iswanto
Computer Science Department
School of Computer Science
Bina Nusantara University
Jakarta, Indonesia
irene.iswanto@binus.ac.id

***Abstract*— Due to its widespread distribution and the anonymity it offers users, hate speech on social media platforms, particularly Twitter, is a major problem. Because of Twitter's echo chamber algorithm and viral nature, hate speech can spread quickly and lead to societal unrest. The use of BERT-based models for hate speech identification in the Indonesian language has been studied in previous research. This study compares how well several pre-trained BERT models, IndoBERT and mBERT, perform in identifying hate speech on Twitter. The methodology includes gathering datasets, preparing the data, and utilizing the proper metrics to assess the models. IndoBERT successfully outperforms mBERT model according to the average of each evaluation metric shown. By carrying out this study, we hope to advance knowledge of techniques for identifying hate speech on Indonesian social media platforms and enhance the use of BERT models for such analysis.**

Keywords—IndoBERT, mBERT, BERT, Twitter, Hate Speech Detection

I. Introduction

Hate speech is a form of public speech that expresses hate and offensive discourse towards a specific group or individuals. Topics commonly used as grounds for hatred include race, religion, gender, and sexual orientation [1]. Nowadays, especially in social media, hate speech almost always contains abusive language. The effects of continuous hate spread can lead to discriminative human behavior, creating a stigma to other readers thinking that there are no consequences in doing so. Hate speech which contains offensive words and phrases often accelerates the development of social conflict which brings out or triggers people's emotion [2]. During these interactions, people tend to get defensive of themselves and aggressive towards others in a form of hate. One of the social media platforms commonly used in that regard is Twitter.

Twitter is a social media platform like any other social media platform, but where twitter differs from other platforms is that a post is called a tweet and a retweet feature to forward it to your followers' feed. Despite its popularity, Twitter is increasingly used to spread hate and misinformation due to its viral nature and anonymity [3]. Twitter allows its users to remain anonymous which makes it easier for the user to spread hate and information without any severe consequences [4]. Twitter's unique retweet feature can make a tweet go viral even if the tweet itself contains negativity. In addition, Twitter's algorithm creates an echo chamber where users are only exposed to information and opinions that align with their existing beliefs. Users commonly use the platform to get updates from their personal and professional connections and to see breaking news from around the world. There are no restrictions on what you can post on Twitter, so you can post freely. As a result, you can easily post negative comments and hate messages on the platform [5]. In an Indonesian Twitter atmosphere, it is common to see arguments in any social media between Indonesians about any given topic. A common occurrence is that we often see news posts and there is no shortage of hate comments in the replies i.e. discrimination. To combat this problem, it is important to filter and identify hate tweets [6].

Bidirectional Encoder Representations from Transformer (BERT) is designed to be trained on large amounts of text data and fine-tuned for various NLP (natural language processing) tasks such as text classification, question answering, and sentiment analysis. was shown. BERT uses a Transformer architecture that can process text bi-directionally [7]. In other words, you can understand the meaning of a word based on the words before and after the sentence. This bidirectional approach makes BERT one of the most advanced NLP models available today, capable of producing state-of-the-art results on a wide range of language understanding tasks. Additionally, there are other language models based on the BERT architecture, such as IndoBERT, DistilBERT. IndoBERT is trained on a large corpus of Indonesian text which means it is more effective to detect Indonesian messages [8].

Based on our findings previous works utilizing BERT models regarding hate speech detection in Twitter is uncommon but not rare, such as paper [9] conducted research comparing performance between IndoLEM and IndoBERT for NLP by performing various tasks which resulted in

IndoBERT achieving better performances over most tasks. Paper [10] conducted research utilizing DistilBERT after using Support Vector Machine (SVM) for classification of hate speech in Indonesian twitter. Lastly paper [11] compares the multilingual transformer model which are mBERT and XLM-R and BERTopic in classifying Indonesian hoax news from news article datasets.

Our paper aims to determine the differences between different pre-trained BERT models in hate speech detection using the Indonesian language. Overall, conducting this research can help improve our understanding of hate speech detection methods in Indonesian social platform and contribute to the development of select BERT models from the analysis done in this research.

This paper is divided into five sections. The second section will be discussing more from previous works. The third section discusses the datasets used and our methods in this research. The fourth section discusses the result of our research. The last section will discuss the conclusion.

II. Literature Review

Hate speech spreading all over social media platforms, Twitter mostly, has been our main problem nowadays as the social media grows. It could be difficult to indicate manually as the massive users of social media have freedom to post anything. There have been several works in literature conducted to automatically detect hate speech or any similar type of language.

To gain at least an output, there must be at least an input as well. In this case, a dataset needs to be processed so that a model could be testified if it is already accurate for addressing certain problems. Most papers have been done using multi-label datasets. Basically, this type of dataset will classify the data into several classes which represent the category of the data themselves. There are a few papers [12], [13], [14] we have found that used multi-label datasets with varying classes for each of them. For instance, papers [12], [13], using Indonesian multi-label hate speech datasets determined the classes in different manners. Paper [12] identified them as "Hate Speech," "non-Hate Speech," "Abusive Language," or "non-Abusive"; paper [13] identified them as "Religion," "Race," "Physical," "Gender," or "Slander"; and paper [14] identified them as "Negative," "Neutral," or "Positive".

In order to process the dataset, most researchers in their studies conducted data preprocessing before processing the dataset into a classification method since usually Twitter messages are unstructured and contain lots of noise affecting the method accuracy used [15]. The data are preprocessed through several steps such as removal of noise characters, normalization, lowercasing, etc. [15]. Only using Twitter message datasets provided on the internet are usually not sufficient for researchers. To solve the problem, the researchers either combine multiple datasets or obtain tweets directly from the Twitter application itself. For instance, study [13] used additional datasets obtained from Twitter using Twitter Search API, namely Tweepy Library.

Hate speech detection or similar topics have been extensively discussed in previous papers in the past few years. Classification models are mostly used to address problems related to detecting hate speech. Those classification models fall into two categories, machine learning models (traditional ones) and deep learning models. Machine learning models have been used to handle hate speech detection in many papers previously, such as SVM [16], Random Forest [17], etc. However, deep learning models have started being used extensively as they perform better than machine learning models. Three popular deep learning models have been used lately, BERT [18], [19], [20], [21], [22], Neural Network (CNN and/or RNN) [23], [1], and LSTM [10].

CNN has been used for handling hate speech detection. Reference [23] conducted 3 experiments using CNN, GRU, and a combination of CNN and GRU to handle Arabic hate speech detection. The dataset used was manually obtained from Twitter using the standard Twitter streaming API, Tweepy. On paper [16], which uses one of the deep learning models CNN to classify their text datasets, has stated that CNN are good at extracting hidden features from text without hand-crafted feature engineering. CNN could also capture semantically equivalent terms using word embeddings. This model can achieve this by using several layers for feature extraction and classification. It is trained using backpropagation and optimized using stochastic gradient descent (SGD) with a cross-entropy loss function.

Another type of deep learning model proposed by paper [15] is a BERT model. The advantage of picking transformer models over other models is that it can capture long-term dependencies which allows for operations that were otherwise not possible in the past. Using the transformer model also allows parallel processing of input features that allows for faster processing. There are many variations of the BERT models, but the one that was proposed in paper [15] is the DistilBERT model. The model uses only half the parameters present in the regular BERT model but is still able to maintain the performance of the BERT model. The DistilBERT model is also able to make inferences with 60% faster speed. Although the comparison made by paper [15] concluded that most transformer models' differences were negligible, the DistilBERT model had the best accuracy at 92% accuracy.

Paper [24] aims to improve the classification of Indonesian hate comments by comparing three models that was pretrained on Indonesian text datasets. The three models are IndoBERT, m-BERT, and Indo RoBERTa small. They used the F1 score evaluation matrix to test the performance of each model. Their conclusions state that in each training set, IndoBERT managed to outperform the other models being compared.

Study [18] utilized BERT with two steps, pre-training and fine-tuning. The authors tried to improve the fine-tuning by testing it into a broad set of NLP tasks. First, the pre-trained parameters were used to initialize the BERT model. The parameters were then fine-tuned by using data from the downstream tasks that were already labeled. Two size models of BERT were used, BERTlarge and BERTbase tested on 4 datasets, such as GLUE, SQuAD v1.1, SQuAD v2.0, and SWAG. Study [20] also utilized BERT, specifically IndoBERTweet, which was used for Indonesian Twitter. Five IndoBERTweet models were used. Either of them was pretrained from scratch, while the rest of them were pretrained based on domain-adaptive pretraining. Those models are tested on several tasks, such as sentiment analysis, emotion classification, hate speech detection, and named entity recognition using a dataset obtained manually from

Twitter. Another example, reference [19] used BERTopic along with back translation and data preprocessing to get the best results in detecting hate speech in Parler. Back translation basically translates the text into five different languages and changes it back to the original one. In Spanish hate speech detection, BERT also might be used as the base model. Study [21] proposed multi-task learning that is expected to improve the performance on each task. To improve the tasks, the model used related tasks, hate speech detection, polarity classification, and emotion classification. BERT encoder was used to allow each task to share its features with each other so that the model can learn what these features are.

The last deep learning model commonly used, LSTM, was utilized to handle Indonesian hate speech detection in the study [25]. That paper used Bi-LSTM architecture with 3 layers, such as an embedding layer, Bi-LSTM, and a fully connected neural network. The paper also used an IndoBERT tokenizer before the dataset was processed in the proposed model.

As elaborated, classifier models fall into two types, which are traditional classifier models and deep learning models. The traditional ones do not require many datasets, while the deep learning ones need lots of datasets in order to get the best output. In this paper, we will be using either one of them. Since we have collected quite numerous datasets, we will be utilizing BERT - one of deep learning models - to gain better outputs than traditional ones. Two BERT models will be compared, which are IndoBERT and m-BERT.

III. Methodology

The methodology proposed will be divided into 4 sections illustrated in the figure below.

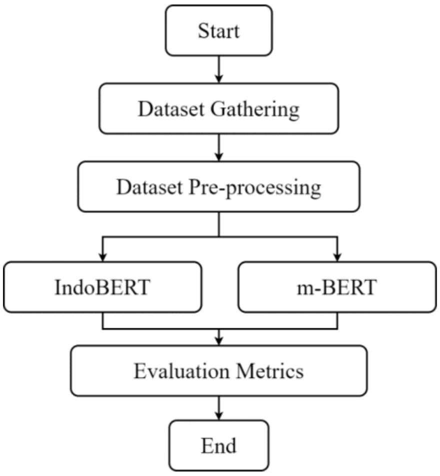

Fig. 1. Methodology Flowchart.

As presented above, we will collect the datasets required from Kaggle. Once collected, they need to be pre-processed first in order to reduce any unnecessary content from the datasets obtained. The pre-processed datasets, then, could be processed by 2 different BERT models. Since BERT itself already has feature extraction embedded in the model itself, there is no need to perform any feature extraction first like some of others do. Lastly, we will conduct an evaluation of the entire chosen models using the metric we choose.

A. *Dataset Gathering*

The datasets used in this paper are the datasets we will collect from Kaggle. We have already collected approximately 13169 tweets data from Kaggle. The data are labeled into multiple labels (12 labels), containing HS, Abusive, HS_Individual, HS_Group, HS_Religion, HS_Race, HS_Physical, HS_Gender, HS_Other, HS_Weak, HS_Moderate, and HS_Strong. A multi-label dataset basically allows a row of data to fall into multiple categories.
Dataset: https://www.kaggle.com/datasets/ilhamfp31/indonesian-abusive-and-hate-speech-twitter-text

B. *Dataset Preprocessing*

Preprocessing datasets is one necessary to do before they can be processed well through the proposed model we will be conducting. To generate the best result, all unnecessary content existing in the entire data must be reduced. We, therefore, will perform a few steps of dataset preprocessing. The steps will be as follows.

1. Turning all words into lowercase.
2. Removing all unnecessary characters (new line, retweet symbol, username, URL, extra spaces).
3. Removing non-alphanumeric characters.
4. Removing stopwords.
5. Spell checking for possible errors.

Since we are using BERT models, some additional steps need to be taken. The steps are as follows.

6. Tokenize the data into subwords that are compatible with the BERT models.
7. Add padding to make the tokenized sequences into the same fixed-length of inputs (BERT models can only accept inputs of a fixed-length).
8. Adding segment IDs to mark which tokens belong to which segment (Used for when more than a single sentence is being inputted or if input consists of multiple parts).
9. Adding attention masks for marking which tokens are considered irrelevant or used for padding and which are considered words and are relevant for the model.

After preprocessing the dataset, it will be split into train, test, and validation dataset in certain ratio automatically determined by the function of the models themselves.

C. *Proposed Method*

BERT models are one of the most used models for NLP. This is largely due to its unique capabilities to solve many long-standing problems in NLP. Some of these solutions are exceptionally useful when trying to detect hate speech from texts. One problem that typically arises is the appearance of out of vocabulary words in the texts that are not present in the language's official vocabulary. BERT models are able to infer the definition of these words using its word to subwords tokenization. Another common issue in other NLP models is their inability to infer the whole sentence's context and is only able to infer the context and meaning of each individual word. BERT models can achieve this as they are considered a bidirectional model. Which means that they will also consider the context of the words to the left and right. Understanding the context of full sentences is crucial because even words with good connotations could still be used as a bad connotation if the sentence is formed as such.

The architecture of BERT models is composed of encoder layers used in transformer architectures. Depending on the variant of the BERT model, the model could have 6, 12, or 24 layers of transformer encoders within its encoder stack. The number of encoders within the encoder stack will affect how well they perform NLP tasks by allowing deeper bidirectional representations.

Fig. 2 shows the number of encoders contained in the encoder stacks of a BERT model. As stated in the previous paragraph, the number of encoders present in an encoder stack depends on the model itself. Encoders used by BERT models are like the regular transformer model's encoders. Each encoder layer has two sub-layers, a multi-head self-attention mechanism and a feed-forward network. The self-attention mechanism's task is to weigh the importance of each word and its contribution in a sentence. Meanwhile, the feed-forward network is used to capture possible complicated patterns that were not originally captured by the self-attention mechanism.

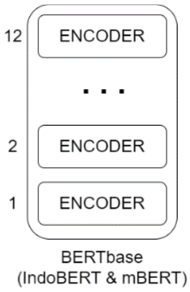

Fig. 2. BERT Models' architecture

The BERT model was pre-trained using two tasks. Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM is a task where a single or multiple tokens are masked or hidden, and the model is tasked with predicting the masked token using the context of all other available tokens to its right and left. Meanwhile, NSP is a task where the model is given two segments of text and the model is asked to determine whether the two texts are part of the same sequence. This allows BERT models to adapt to new vocabulary even without knowing the new word's context. It will also allow better fine-tuning for more specific tasks.

For this research we have chosen two variants of the BERT models that can classify Indonesian texts. The two models are IndoBERT and m-BERT. IndoBERT was pre-trained with exclusively Indonesian text datasets, while m-BERT is a multilingual BERT model pre-trained with several languages including the Indonesian language. In terms of architecture, m-BERT's architecture added a few small extra modifications to BERT's original architecture to better handle multilingual text processing.

D. Experimental Setup

TABLE I. THREE DIFFERENT VALUES USED FOR TESTING THE HYPERPARAMETERS

	Learning Rate	Batch Size	Epoch
Low	2e-3	4	2
Medium (Default)	2e-5	8	5
High	2e-7	16	10

TABLE II. SEVEN EXPERIMENTS CONDUCTED FOR BOTH MODELS (INDOBERT AND MBERT)

	Learning Rate	Epoch	Batch Size
Experiment 1	Medium (Default)	Medium (Default)	Medium (Default)
Experiment 2	Low	Medium (Default)	Medium (Default)
Experiment 3	High	Medium (Default)	Medium (Default)
Experiment 4	Medium (Default)	Low	Medium (Default)
Experiment 5	Medium (Default)	High	Medium (Default)
Experiment 6	Medium (Default)	Medium (Default)	Low
Experiment 7	Medium (Default)	Medium (Default)	High

To identify what independent variables affect the inference results and performance of each of the models, an experimental setup was made to allow for changes within the hyperparameters to be made. The hyperparameters that will be modified are the model's learning-rate, training epoch, and training batch size. Each parameter will be tested with three different values. These values being "Low", "Medium", and "High" category values.

When conducting hyperparameter testing, only the values of the hyperparameter being tested are changed. This allows for a controlled experiment where the effects of the changes in the tested hyperparameters can be accurately measured. All other hyperparameters will be set to their default values to ensure that the results are not influenced by any other hyperparameters.

E. Evaluation Metrics

To know how good the two models we will be using are, an evaluation must be conducted. We decided to use three of the existing evaluation metrics, which are the F-1 score, ROC-AUC, and accuracy score. Sklearn library will be utilized for measuring those three evaluation metrics using Python programming language. Once the two BERT models are evaluated, we will compare all those two models and make a conclusion based on the evaluation.

IV. RESULT AND DISCUSSION

The results and discussion section will show the results of the proposed models' inferences. There are fourteen experiments that have been conducted to test the effectiveness of the two models we have chosen. Each model was tested through seven experiments. In the first experiment, all of the three hyperparameters were set to medium values. In the second and third experiment, the learning rate values were set to low and high values respectively, and so were the rest hyperparameters in the remaining experiment. The results of each experiment on each model are presented in two tables below.

TABLE III. Experiments on IndoBERT Models

	F1 Score	ROC AUC	Accuracy Score
Experiment 1	**0.98**	**0.96**	**0.88**
Experiment 2	0.89	0.93	0.81
Experiment 3	0.91	0.94	0.84
Experiment 4	0.91	0.95	0.84
Experiment 5	0.92	0.95	0.86
Experiment 6	0.89	0.93	0.82
Experiment 7	0.89	0.93	0.81
Average	0.91	0.94	0.84

TABLE IV. Experiments on mBERT Models

	F1 Score	ROC AUC	Accuracy Score
Experiment 1	0.84	0.89	0.75
Experiment 2	0.76	0.85	0.66
Experiment 3	**0.86**	**0.9**	**0.77**
Experiment 4	0.85	**0.9**	0.76
Experiment 5	0.69	0.79	0.61
Experiment 6	0.69	0.79	0.6
Experiment 7	0.59	0.74	0.48
Average	0.75	0.84	0.66

The results of modifying each of the specified hyperparameters are shown in TABLE III and TABLE IV. The default values or medium values used are learning-rate: 2e-5, batch-size: 8, epoch: 5. The results of using these values are shown in TABLE III (Experiment 1) for the IndoBERT model and TABLE IV (Experiment 1) for the mBERT model.

By modifying the learning-rate to its high value of 2e-7 and its low value of 2e-3, its can be seen that the medium learning-rate value of 2e-5 possess the best performance in both models according to its F1, ROC. Using the high and low values causes a drop in accuracy in the model inferences. This is shown in TABLE III (Experiment 2 and 3) and TABLE IV (Experiment 2 and 3). This drop in accuracy could cause by an unstable learning process due to a high learning-rate or a sub-optimal training process.

Modifying the epochs to a low value of 2 epochs and a high value of 10 epochs gives varying accuracy. Using the medium or default values of 5 epoch still has the best accuracy out of the other values for both models according to its F1, ROC, and accuracy. This can be seen in TABLE III (Experiment 4 and 5) and TABLE IV (Experiment 4 and 5). The drop in accuracy may be caused by overfitting for having too much epoch or underfitting from having too little epochs.

Lastly, the experiment modified the batch sizes of each model to its low value of 4 and high value of 16. Like previous hyperparameters, using the medium or default value of 8 batch size for the batch sizes proves to be the most accurate. This can be seen in This can be seen in TABLE III (Experiment 6 and 7) and TABLE IV (Experiment 6 and 7). Using more batch size should result in a more accurate model, but more batch sizes require more epochs to converge on the minimum validation loss. Since in this experiment we exclusively modify the batch sizes only and not the batch sizes and the epochs simultaneously, it could be the cause for accuracy dropping.

V. Conclusion

In this paper, we investigated the effectiveness of two BERT models, IndoBERT and m-BERT, for the task of multi-label hate speech detection in Indonesian Twitter. We conducted a comprehensive literature review on the existing methods and datasets for hate speech detection, and proposed a methodology that consists of data collection, preprocessing, model training, and evaluation. We performed 14 experiments with different hyperparameters for each model, and evaluated them using F1 score, ROC AUC, and accuracy score. Our results showed that IndoBERT consistently outperformed m-BERT across all metrics, indicating that IndoBERT is more suitable for Indonesian hate speech detection. We also experimented with the various hyperparameters that can affect the NLP model's inferences and found that the perfect value for hyperparameter can vary depending on the factors, parameters, and dataset. We discussed the possible factors that contributed to the performance gap between the models, and acknowledged the limitations of our study.

References

[1] M. O. Ibrohim and I. Budi, “Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter,” 2019. doi: 10.18653/v1/w19-3506.

[2] M. O. Ibrohim and I. Budi, “A Dataset and Preliminaries Study for Abusive Language Detection in Indonesian Social Media,” in *Procedia Computer Science*, 2018. doi: 10.1016/j.procs.2018.08.169.

[3] R. T. Mutanga, N. Naicker, and O. O. Olugbara, “Hate speech detection in twitter using transformer methods,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020, doi: 10.14569/IJACSA.2020.0110972.

[4] M. A. Fauzi and A. Yuniarti, “Ensemble method for indonesian twitter hate speech detection,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 11, no. 1, 2018, doi: 10.11591/ijeecs.v11.i1.pp294-299.

[5] P. K. Roy, A. K. Tripathy, T. K. Das, and X. Z. Gao, “A framework for hate speech detection using deep convolutional neural network,” *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3037073.

[6] W. Chen and W. Fu, “Negative information filtering algorithm based on text content in multimedia networks,” *International Journal of Performability Engineering*, vol. 15, no. 11, 2019, doi: 10.23940/ijpe.19.11.p26.30613071.

[7] G. Z. Nabiilah, S. Y. Prasetyo, Z. N. Izdihar, and A. S. Girsang, “BERT base model for toxic comment analysis on Indonesian social media,” *Procedia Comput Sci*, vol. 216, pp. 714–721, 2023, doi: https://doi.org/10.1016/j.procs.2022.12.188.

[8] F. Ihsan, I. Iskandar, N. S. Harahap, and S. Agustian, “Decision tree algorithm for multi-label hate speech and abusive language detection in Indonesian Twitter,” *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 4, pp. 199–204, Oct. 2021, doi: 10.14710/jtsiskom.2021.13907.

[9] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, “IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP,” in *Proceedings of the 28th International Conference on Computational Linguistics*, Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 757–770. doi: 10.18653/v1/2020.coling-main.66.

[10] K. M. Hana, Adiwijaya, S. Al Faraby, and A. Bramantoro, “Multi-label Classification of Indonesian Hate Speech on Twitter Using Support Vector Machines,” in *2020 International Conference on Data Science and Its Applications (ICoDSA)*, IEEE, Aug. 2020, pp. 1–7. doi: 10.1109/ICoDSA50139.2020.9212992.

[11] L. B. Hutama and D. Suhartono, “Indonesian Hoax News Classification with Multilingual Transformer Model and

BERTopic," *Informatica*, vol. 46, no. 8, Nov. 2022, doi: 10.31449/inf.v46i8.4336.

[12] G. B. Herwanto, A. M. Ningtyas, I. G. Mujiyatna, K. E. Nugraha, and I. N. Prayana Trisna, "Hate Speech Detection in Indonesian Twitter using Contextual Embedding Approach," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 15, no. 2, 2021, doi: 10.22146/ijccs.64916.

[13] M. O. Ibrohim and I. Budi, "Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter," 2019. doi: 10.18653/v1/w19-3506.

[14] Yuyun, Nurul Hidayah, and Supriadi Sahibu, "Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 4, 2021, doi: 10.29207/resti.v5i4.3146.

[15] R. T. Mutanga, N. Naicker, and O. O. Olugbara, "Hate speech detection in twitter using transformer methods," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 9, 2020, doi: 10.14569/IJACSA.2020.0110972.

[16] P. K. Roy, A. K. Tripathy, T. K. Das, and X. Z. Gao, "A framework for hate speech detection using deep convolutional neural network," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3037073.

[17] Y. Tang and N. Dalzell, "Classifying Hate Speech Using a Two-Layer Model," *Statistics and Public Policy*, vol. 6, no. 1, 2019, doi: 10.1080/2330443X.2019.1660285.

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Oct. 2018, [Online]. Available: http://arxiv.org/abs/1810.04805

[19] N. Schneider, S. Shouei, S. Ghantous, and E. Feldman, "Hate Speech Targets Detection in Parler using BERT," Apr. 2023, [Online]. Available: http://arxiv.org/abs/2304.01179

[20] F. Koto, J. H. Lau, and T. Baldwin, "IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization," Sep. 2021, [Online]. Available: http://arxiv.org/abs/2109.04607

[21] F. M. Plaza-Del-Arco, M. D. Molina-Gonzalez, L. A. Urena-Lopez, and M. T. Martin-Valdivia, "A multi-task learning approach to hate speech detection leveraging sentiment analysis," *IEEE Access*, vol. 9, pp. 112478–112489, 2021, doi: 10.1109/ACCESS.2021.3103697.

[22] X. Liu, H. Lu, and A. Nayak, "A Spam Transformer Model for SMS Spam Detection," *IEEE Access*, vol. 9, pp. 80253–80263, 2021, doi: 10.1109/ACCESS.2021.3081479.

[23] R. Alshalan and H. Al-Khalifa, "A deep learning approach for automatic hate speech detection in the saudi twittersphere," *Applied Sciences (Switzerland)*, vol. 10, no. 23, pp. 1–16, Dec. 2020, doi: 10.3390/app10238614.

[24] F. Ihsan, I. Iskandar, N. S. Harahap, and S. Agustian, "Decision tree algorithm for multi-label hate speech and abusive language detection in Indonesian Twitter," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 4, pp. 199–204, Oct. 2021, doi: 10.14710/jtsiskom.2021.13907.

[25] A. Perwira Joan Dwitama, D. Hatta Fudholi, and S. Hidayat, "Indonesian Hate Speech Detection Using Bidirectional Long Short-Term Memory (Bi-LSTM)," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 2, pp. 302–309, Mar. 2023, doi: 10.29207/resti.v7i2.4642.