JOIV : Int. J. Inform. Visualization, 7(3) - September 2023 773-780

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Indonesian Hate Speech Detection Using IndoBERTweet and BiLSTM on Twitter

Juanietto Forry Kusuma [a,*], Andry Chowanda [b]

[a] BINUS Graduate Program – Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia
[b] Computer Science Department – School of Computer Science, Bina Nusantara University, Jakarta, Indonesia
Corresponding author: [*]juanietto.kusuma@binus.ac.id

***Abstract*— Hate speech is an act of speech to spread hate to other people. In this digital era where everyone connects with social media, hate speech is growing rapidly and uncontrollably. Many people do not realize they are giving hate speech when critics something on social media due to a lack of awareness of the difference between hate speech and free speech. The results make victims feel alienated from society, and the people who spread it would often face the law. Detection in the sentences to identify whether it contains hate speech is essential to counter people's ignorance. For detecting such sentences, a machine learning algorithm is widely used to help identify each sentence. In this paper, we used a subset from machine learning named deep learning with the latest IndoBERT model named IndoBERTweet and combined it with RNN layer named BiLSTM. The appearance of IndoBERTweet opened more chances to further improve text classification performance with the addition of BiLSTM layer. The model first made a token representative from the sentence, then calculated it to analyze and made the classification based on the calculation. For this model to be effective, we trained our model with the labeled public dataset retrieved from Twitter. These datasets are classified into hate speech and non-hate speech, and these labels are applied to the models. We evaluated our model and achieved an accuracy of 93.7%, an improvement for classifying hate speech sentences from previous research.**

***Keywords*— Hate speech; IndoBERTweet; BiLSTM; text classification.**

Manuscript received 20 Jul. 2022; revised 20 Jan. 2023; accepted 5 Feb. 2023. Date of publication 10 Sep. 2023.

I. INTRODUCTION

The use of internet technology in human life has become a necessity today. This technology is mainly used to access and find the information needed. The internet can also connect people from around the world using social media. The use of social media is common, and almost everyone has a social media account. There has been an increase in active users of social media by 13.2% since 2020 [1]. The pandemic situation that happened in 2020 had a significant impact on increasing social media users. This proves that social media has become a place where many people connect and interact with one another. Almost everyone has a social media account like Facebook, Twitter, and Instagram. This high number of social media accounts led to many fake accounts that have been used with ill intentions. Those fake accounts were used to attack persons or organizations just for self-satisfactory. With the ease of creating social media account that only takes a few minutes, they can easily launch attacks. One of its common attacks is hate speech. Hate speech is an utterance that explicitly attacks a person or group based on ethnicity, race, and religion [2]. Generally, those who carry out this action dislike other people or groups. The purpose of hate speech varies depending on the person's intentions but aims typically to disturb, threaten, slander, and incite [3]. This action is made more accessible with the presence of social media.

The spread of hate speech on social media significantly impacts society. This is very troubling to some victims who are likely to form a suspicion of a particular community. Victims who often received hate speech would isolate themselves and add more prejudice towards them. Society's perspective has been manipulated by hate speech content on social media. The spreading of this content is massive and structured, causing this content to spread faster [4]. In Indonesia, this content received a significantly increased in 2017. Because of the alleged case of blasphemy committed by the governor of Jakarta, this topic is often discussed on social media. This case began when someone edited the video in which the governor was making a speech and then added a text saying that this governor was blasphemy. This concerns many people because it is easy to record, spread, and make

narration to lead to a particular opinion. Hate speech, such as insults, slander, blasphemy, dishonorable actions, provocations, and spreading false news, causes a lot of negative things [5].

Many people do not realize that the difference between hate speech and free speech is one factor that helps spread hate speech. Expressing an opinion is the right of every person, but this freedom has limitations. Rules and laws are made to identify expression, which aims to make walls between free speech and hate speech [6]. To achieve this, support factors and systems are needed to help distinguish between hate speech and not. Many researchers have studied to create a system that can help distinguish an opinion, including hate or not. However, due to a large number of hate speech widely spread on social media, further research is needed to get a system that can detect it accurately and quickly.

Many researchers have developed a system to detect hate speech on social media. Mozafari, Farahbakhsh and Crespi [7] conducted research using Bidirectional Encoder Representations from Transformer (BERT), one of the deep learning models, combined with the Bidirectional Long Short-Term Memory (BiLSTM) model and the Convolutional Neural Network (CNN) model to detect hate speech in the English language. From the results of their research, it is known that the BERT model combined with other deep learning models, especially the Natural Language Processing (NLP) model, performs better. In Indonesia, researchers such as Herwanto *et al*. [8] have detected hate speech on Twitter datasets using the Gated Recurrent Unit (GRU) model and word embedding and contextual embedding. Their research used two datasets. Each performed differently in each model. Another research in Indonesian uses the Indonesian BERT model (IndoBERT), conducted by Koto, Lau, and Baldwin [9] in his research. He developed using the IndoBERT model by training using additive domain-specific vocabulary, resulting in a new IndoBERT model called IndoBERTweet. This model was trained with Indonesian tweets obtained using Twitter API and got 409M word tokens, two times bigger than the word tokens used in the training data for IndoBERT. Hate speech detection research in the Indonesian language using deep learning is not much compared to research in another language. Researchers faced this mainly because only a few Indonesian datasets are available to the public. Because of that, many researchers collect their data with Twitter API (Application Programming Interface) for data crawling. The other problem is that many words used are not ordinary words. Therefore, researchers must add another work to their research to translate those words.

The research on hate speech detection in English produced many outstanding performances, especially using the deep learning model. Those great performance models open a chance to implement the deep learning model for Indonesian hate speech detection. The state-of-the-art model like BERT and combining BERT with other deep learning classifier models resulted in a more significant performance. The result of previous research, especially in the Indonesian language, shows room for another improvement in classification. The appearance of the newer IndoBERT named IndoBERTweet means more opportunities to increase the text classification performance further. Previous research shows that adding Recurrent Neural Network (RNN) after the Transformer model could achieve a better result in performance. Therefore, this research proposed the combined model of the Transformer model named IndoBERTweet, the state-of-the-art IndoBERT, and RNN model named BiLSTM for hate speech detection in the Indonesian language. Using two publicly Indonesian datasets by Alfina *et al*. [10] and Ibrohim and Budi [11] for hate speech to achieve the best performance with the proposed model. With this research, we aim to improve the performance of hate speech classifications in Indonesia using IndoBERTweet combined with the BiLSTM model.

II. Material and Method

In this section, we present the related research and methods for research. For the method, the first was defining the dataset and then cleaning the dataset in preprocessing. The following process is building the model architecture and the evaluation. The following subchapter will explain each process in detail.

A. Related Works

Researchers have done considerable research regarding hate speech detection. Vigna *et al*. [12] proposed two machine learning models, Support Vector Machine (SVM) and Long Short-Term Memory (LSTM). Their research experimented with two types of classifiers: the first is three labels (strong hate, weak hate, and no hate), and the second is two labels (hate and no hate). Their experiment resulted in SVM with two labels gaining better accuracy than other models in their research. Gambäck and Sikdar [13], in their research for classifying hate speech using CNN with random vector, word2vec, n-grams, and character n-grams. Their research found that CNN with word2vec gained the best score in hate speech detection using English. Another research by Al-Makhadmeh and Tolba proposed a deep learning model called Killer Natural Language Processing Ensemble Deep Neural Network (KNLPEDNN). This model achieved higher accuracy than another model they tested [14]. Faris et al. [15] used the Arabian dataset to experiment with different combinations of word embedding and deep learning. Based on their research, the best performance was achieved by Aravec word embedding with N-gram and Skip-gram. Kapil, Ekbal, and Das [16] used four deep learning models with different word vectors resulting in the best performance achieved with the BiLSTM model with GloVe combined with Character-CNN. Nguyen et al. experimented with five deep learning models in a different language like Vietnam. Those models are TextCNN, Very Deep Convolutional Neural Network (VDCNN), BiLSTM, LSTM + CNN, and Spatiotemporal Attention Recurrent Neural Network (SARNN) [17]. SARNN with comment_tokenize as word embedding between these models achieved a more excellent F1 score than other models. The researchers mainly focused on one language, but Aluru *et al*. took a different approach. In their research, Aluru *et al*. [18] experimented with four deep learning models and used sixteen datasets from nine different languages, one of which iIndonesian. They used the Multilingual BERT (mBERT) model to handle datasets from non-English languages. For training the model, they used two different methods: training the model with the same language as in testing data and training the model with a different language. This different training method resulted in the model

achieving more excellent performance and training with the same language. Sutejo and Lestari did another research using an Indonesian dataset, and they used the LSTM model to experiment with different datasets like text and audio with outstanding performance [19].

Another researcher who performed outstandingly in their research is Isnain, Sihabuddin, and Suyanto [20]. The study proposed the BiLSTM model and Word2Vec with Continuous Bag of Words (CBOW) architecture to classify hate speech in the Indonesian language. They could achieve a great result by tuning some of its parameters like learning rate, epoch, and the number of neurons on the hidden layer. Research on hate speech using the IndoBERT model was used by Koto, Lau, and Baldwin [9]. They proposed the newer IndoBERT model called IndoBERTweet, trained with additive domain-specific vocabulary using the Indonesian tweets dataset they gained from Twitter API. Their research used the model to perform several classification tasks, including hate speech detection. The result of hate speech detection shows that their new model achieved better results than the IndoBERT model, with 86.1%. Marpaung, Rismala, and Nurrahmi [21] proposed a combined model consisting of IndoBERT and Bidirectional GRU (BiGRU) as the classifier achieved an accuracy of 84.7%.

Based on previous research, the research regarding hate speech has improved so much in recent years. This could happen because there are many publicly English datasets for researchers to use in their research. However, this is not the case with other non-English languages, as there are few datasets compared to English. Indeed, as time goes on, the different languages will have some reliable public datasets. Research depends on supervised learning, so some datasets that are not labeled must be labeled. Labeling the dataset takes so much time and effort as it needs professionals who objectively identify every sentence. If the dataset is extensive, it requires so many people to do the job. The research on hate speech detection in Indonesia could still be developed further. New deep learning models such as the IndoBERTweet have emerged in recent years and can open up opportunities for further research.

In this paper, we only use two datasets that are currently available as public datasets, and other researchers also use these two datasets. The performance of deep learning models in previous research, especially those combined, almost always had an excellent performance. We, therefore, used the combined deep learning model in this research to see if it performs as well as in the English language. As in Indonesian, combining IndoBERT and BiGRU proved to be effective in improving classification performance [21]. For this research, we are using IndoBERTweet and BiLSTM because, as in previous research, combining transformer models such as IndoBERT and RNN layer proved to increase the performance; therefore, we use IndoBERTweet, which is the next improvement from IndoBERT combined with BiLSTM layer. Also, in previous research, many researchers still use IndoBERT as the transformer model, while IndoBERTweet is the first large-scale pre-trained model with additive domain-specific vocabulary and has more training data than IndoBERT [9]. Thus, in this research, we used IndoBERTweet as our transformer model. This research is the first research using IndoBERTweet combined with BiLSTM for hate speech classification. We also fine-tuned the pre-trained IndoBERTweet model and evaluated the performance with our combined models.

B. Dataset

The dataset uses the Indonesian public dataset from Alfina *et al*. [10] and Ibrohim and Budi [11]. These two datasets were made by crawling data using Twitter API. The first dataset from Alfina *et al*. is a two-label dataset: Hate Speech (HS) and non-Hate Speech (non-HS), containing 713 data. The other dataset is multi-labels totaling twelve broken down into HS, HS_individual, HS_Group, HS_Religion, HS_Race, HS_Physical, HS_Gender, HS_Other, HS_Weak, HS_Moderate, HS_Strong, and abusive, contains 13,169 data. For this research, this multi labels dataset is merged into two labels consisting of HS and non-HS. Table I shows examples of both labels in both datasets. Fig. 1 shows how frequently each word in both datasets uses the word cloud.

TABLE I
TWEET AND LABEL EXAMPLE FROM THE DATASET

Dataset	Tweet	Label
Ibrohim & Budi [11]	di saat semua cowok berusaha melacak perhatian gue kamu lantas remehkan perhatian yang gue kasih khusus ke kamu basic kamu cowok bego (when all the guys trying to catch my attention you just underestimate my special care for you, you are a stupid boy)	HS
	siapa yang telat memberi tau kamu sarap gue bergaul dengan cigax jifla calis sama siapa itu licew juga (who was late to tell you, I hang out with cigax jifla calis and licew too)	Non-HS
	salah jokowi ahok kafir ateis (Jokowi and ahok is wrong they are atheist and infidel)	HS
Alfina et.al. [10]	fadli zon menteri dalam negeri menonaktifkan ahok gubernur daerah khusus ibu kota (fadli zon minister of home affairs deactivates ahok, governor of the special capital region)	Non-HS

Fig. 1 Word Cloud of the two datasets

C. Preprocessing

The purpose of preprocessing the data is to clean the noise by removing and normalizing unnecessary words. This process concludes lowercase, removing retweet (RT) words, Twitter usernames, URL, emoji, and excessive spacing and line break use. We also used a dictionary provided by the Ibrohim dataset to convert the slang and misspelled words back to their original form. Finally, the dataset was split into training, validating, and testing data for the last part. The ratio of splitting data is 80% training and 20% testing. Then the training data was divided again into 20% for validating data

and the remaining 80% for training. In our research, we did not use stop word removal as it would decrease the performance of our model [21].

D. Model Architecture

In this research, we constructed three model architectures. The first is our proposed model, and the other two are models for comparison to our proposed one. The other two models are the IndoBERTweet and CNN layer and fine-tuned IndoBERTweet model. IndoBERTweet is a transformer model with 12 hidden layers, 12 attention heads, and three feed-forward hidden layers. This research used the pre-trained IndoBERTweet model for our feature extraction process. Before we input the data to the model, the data needed to be encoded using BertTokenizer from Hugging Face. The output from BertTokenizer consists of input ids and an attention mask. We only used the input ids as our input data for our research. Input ids are token indices, numerical representations of tokens building the sequences used as input by the model. Later, these input ids were converted into the vector using Tensorflow Dataset and its respective labels. Table II shows the example of input IDs.

TABLE II
EXAMPLE OF INPUT IDS FROM BERTTOKENIZER

Sequence Input	di saat semua cowok berusaha melacak perhatian gue kamu lantas remehkan perhatian yang gue kasih khusus ke kamu basic kamu cowok bego (when all the guys trying to catch my attention you just underestimate my special care for you, you are a stupid boy)
Tokenize	['di', 'saat', 'semua', 'cowok', 'berusaha', 'melacak', 'perhatian', 'gue', 'kamu', 'lantas', 'remehkan', 'perhatian', 'yang', 'gue', 'kasih', 'khusus', 'ke', 'kamu', 'basic', 'kamu', 'cowok', 'bego'] (when, all, the, guys, trying, to, catch, my, attention, you, just, underestimate, my, special, care, for, you, you, are, a, stupid, boy)
Tokenizer output as input ids	[3, 1485, 1759, 2014, 17257, 3519, 19154, 4082, 9875, 3162, 6849, 30281, 4082, 1497, 9875, 3774, 2523, 1500, 3162, 19723, 3162, 17257, 11619, 4]

For another fine-tuned IndoBERTweet model, we used both input ids and an attention mask from BertTokenizer. The attention mask is used as a binary tensor indicating the position of the padded indices so that the model does not attend to them. Hyperparameter for our fine-tuned model using Adam Optimizer with a learning rate of 1e-5, batch size of 5, and maximum token length of 128. We used a loss function for binary classification named Binary Cross Entropy to evaluate our model since our class only consists of hate speech and non-hate speech.

BiLSTM is an independent two-LSTM architecture with different directions [22]. With these two directions, this model processes input from backwards and forward. One thing that differs from traditional LSTM is that when moving backward, the model stored information from the front, and with two hidden states combined, it could store the information whenever from the back or the front. We used the BiLSTM model as our classifier and received inputs from the last hidden states from IndoBERTweet.

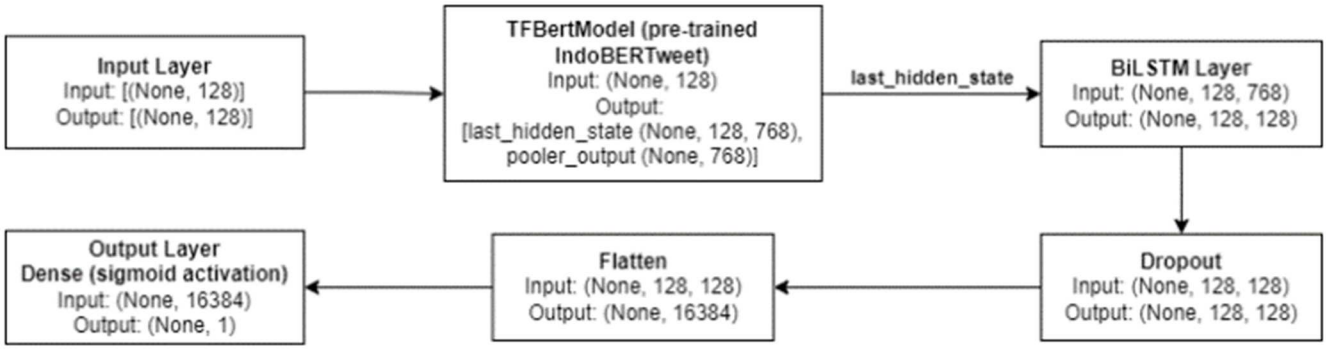

Fig. 2 IndoBERTweet + BiLSTM model

Fig. 2 describes our proposed model, and we used pre-trained IndoBERTweet as the first layer. This layer output tuple of 2 tensors comprises the last hidden state and pooler output. The next layer is the BiLSTM layer, with 64 LSTM units. After the BiLSTM layer, we used the Dropout layer with dropout rates of 0.3 to reduce overfitting when training the model. Finally, we used a flattened layer followed by a Dense layer with sigmoid activation.

The other combined model we constructed was changing the BiLSTM layer with the CNN layer. We used CNN layer because, in many research, CNN proved to improve model performance [7], [13], [16]. In addition, for the fine-tuned IndoBERTweet model, we added a dropout and linear layer where the output feature size is 2. We constructed these two models to compare the proposed model with these two models and to find out if adding an RNN layer such as BiLSTM would improve the performance.

E. Evaluation

For evaluation, we used a confusion matrix to compare the performance of our three models consisting of IndoBERTweet + BiLSTM, fine-tuned IndoBERTweet, and IndoBERTweet + CNN model. We obtained the model's accuracy, recall, precision, and F1 score with the confusion matrix. The confusion matrix is arranged on a 2x2 matrix. The prediction class is arranged horizontally in a row, and the true class is arranged vertically in a column [23]. We can calculate the accuracy, recall, precision, and F1 score from the confusion matrix. The formula can be seen in Eq. (1) to (4).

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$Accuracy = \frac{TN+TP}{TN+FP+TP+FN} \quad (3)$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Notes: TP is True Positive, *FP* is False Positive, *FN* is False Negative, and *TN* is True Negative.

Besides the confusion matrix, we also compare our models with previous models in related fields. The studies that we compared are Marpaung, Rismala, and Nurrahmi [21] and Koto, Lau, and Baldwin [9]. We compared previous research to see if our proposed model could achieve better results than previous works.

III. Results and Discussion

Our models were trained using two datasets. For training our combined model, we used the same learning rate of 1e-5. In the BiLSTM layer, we used a batch size of 10 and 5 epochs for both datasets. We used a batch size of 5 and 8 epochs in the CNN layer for training. The maximum token length is 128 for both datasets. Our research achieved the best performance by combining the model with the BiLSTM layer with an accuracy of 88.6% for the first dataset and 93.7% for the second dataset.

A. IndoBERTweet + BiLSTM

Based on Table III, the confusion matrix gained from the prediction done by the model shows that the model does a lot of misclassifications in the HS class. However, in non-HS classes, the model only does a little misclassification. This could happen because the data used for training is unbalanced, whereas the data in non-HS classes have more than in HS classes. Therefore, the model could perform better in one class and not the other.

TABLE III
CONFUSION MATRIX FROM INDOBERTWEET + BILSTM USING DATASET [10]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	50	3
	Non-HS	6	84

Table IV shows the confusion matrix from the model prediction using another dataset. The other dataset suffered the same problem as the previous one, where the data was imbalanced. At first glance, the misclassification in non-HS looks much more than HS, but there is a slight difference if we calculate the percentage with the amount of all data by each class. The misclassification in HS classes is still higher compared to non-HS by percentage. Unlike the previous dataset, there is only a slight difference between the HS and non-HS.

TABLE IV
CONFUSION MATRIX FROM INDOBERTWEET + BILSTM USING DATASET [11]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	963	175
	Non-HS	125	1371

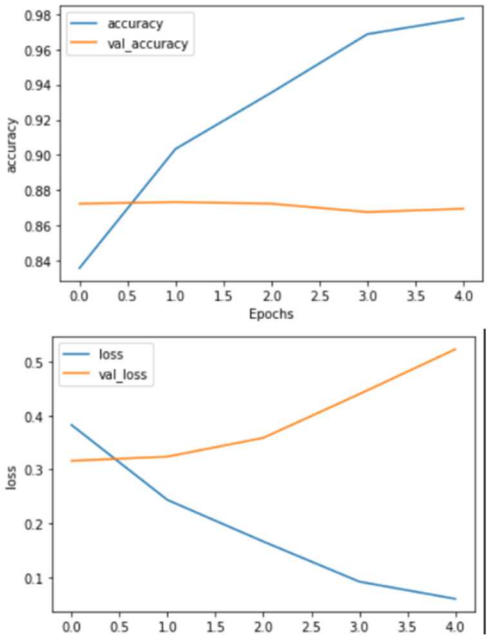

Fig. 3 Accuracy and loss graphic IndoBERTweet + BiLSTM when using dataset Ibrohim and Budi [11]

The result of model accuracy and loss when training and validating is illustrated in fig. 3. The blue line stands for model performance when training, and the orange line stands for validation performance. Our model started with lower accuracy and steadily increased compared to when in validation when in training. Accuracy is stagnant and dropping at epoch three but increases again only at the same level as before dropping. This could happen because our model has a massive number of parameters, and the data for the training process is too small. While calculating the loss from training and validating, the training loss starts at a higher loss than the validation loss. The training loss gradually decreased as the epoch increased, but the validation loss increased. This sign of difference told us that our model is overfitting, so we used a few epochs.

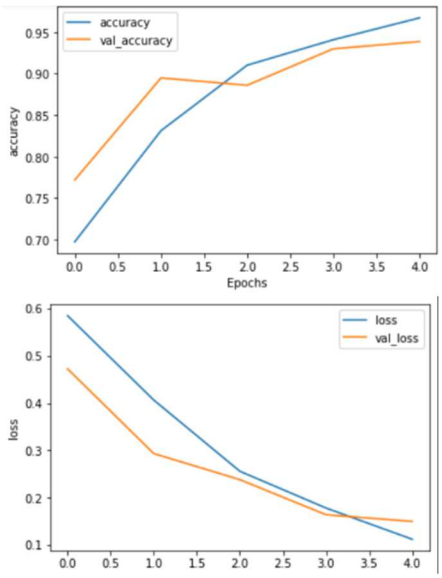

Fig. 4 Accuracy and loss graphic IndoBERTweet + BiLSTM when using dataset Alfina et al. [10]

Fig. 4 shows the result of the model using another dataset. The result from the first epoch is the same as using the first dataset in that the validation performance is higher than training. Unlike in Fig. 3, Fig. 4 shows that accuracy performance in validation and training gradually increases, and at epoch three, the training accuracy is higher than validation accuracy. While the training performance is increasing, the validation accuracy drops at epoch three but then increases again in the next epoch. The training loss also starts at a higher loss compared to the validation loss. As the training goes on, the loss keeps decreasing, and at the final epoch, the validation loss is at a stagnant level where it is only decreased a bit from the previous epoch.

B. IndoBERTweet + CNN

From the results in Table V, the model misclassified class non-HS more than the previous model. However, while the model made more mistakes, and when we calculated the percentage of mistakes from both classes, the model still performed lower in the HS class compared to non-HS.

TABLE V
CONFUSION MATRIX FROM INDOBERTWEET + CNN USING DATASET [10]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	50	8
	Non-HS	6	79

Table VI shows that the model made more mistakes in classification in HS class when predicting using other datasets. The model faces the same problem as in the previous one, where the model performs better in one class. Although at first look, the misclassification in non-HS is more than HS, if we calculate in percentage, the HS still has the most mislabelled data when predicting models.

TABLE VI
CONFUSION MATRIX FROM INDOBERTWEET + CNN USING DATASET [11]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	941	169
	Non-HS	147	1377

The model accuracy and loss when training the model and evaluating the model using validation data for both datasets are illustrated in Fig. 5 and Fig. 6. In Fig. 5, the first epoch resulted in evaluation performance is higher than in training. However, when it goes to the next epoch, the validation performance matched the training. The next epoch until the last epoch shows that while in training, the model performance gradually gets better and better, but in evaluation, the model performance could not keep up with the training performance. The model accuracy throughout the evaluation could not increase; instead, it stays on the same value, drops, and then rise again to the same value. In the loss calculation, evaluating the model keeps increasing until epoch seven and then decreases at the last epoch.

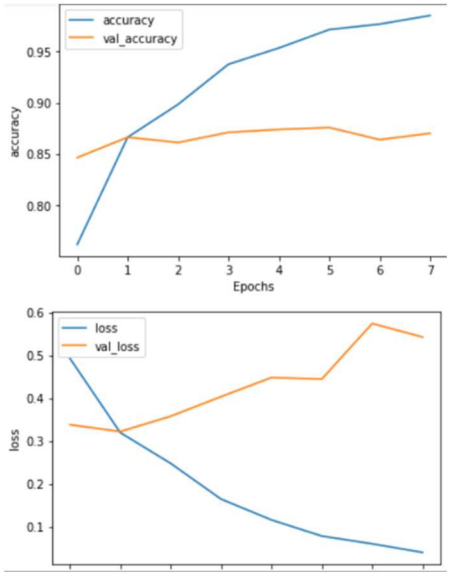

Fig. 5 Accuracy and loss graphic IndoBERTweet + CNN when using dataset Ibrohim & Budi [11]

Fig. 6 shows the model performance using other datasets. Unlike the previous dataset, where the accuracy stays at the same value in evaluating the model, the validation accuracy increases along with the training accuracy. Same to the accuracy, the loss calculations decrease when training and evaluating the model. However, it goes well; at epoch three, the evaluation model performance decreases but gradually increases until the last epoch.

C. Fine-tuned IndoBERTweet

Based on Table VII, the confusion matrix shows that the model makes fewer mistakes when classifying the HS data than the previous two models. However, the model still performs better when classifying non-HS data. Table VIII shows the confusion matrix when using the other dataset. Unlike in IndoBERTweet + CNN model, the fine-tuned IndoBERTweet has fewer misclassification data in both HS and non-HS. Same as the other two models, this model performs better at classifying non-HS data but not as well when classifying HS data.

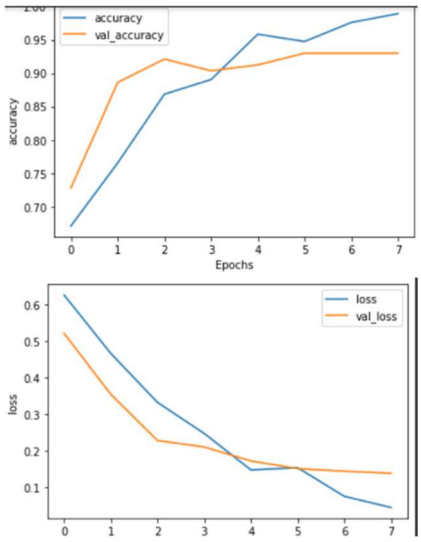

Fig. 6 Accuracy and loss graphic IndoBERTweet + CNN when using dataset Alfina et al. [10]

TABLE VII
CONFUSION MATRIX FROM FINE-TUNED INDOBERTWEET USING DATASET [10]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	51	7
	Non-HS	5	80

TABLE VIII
CONFUSION MATRIX FROM FINE-TUNED INDOBERTWEET USING DATASET [11]

		Actual Class	
		HS	**Non-HS**
Predicted Class	**HS**	951	167
	Non-HS	137	1379

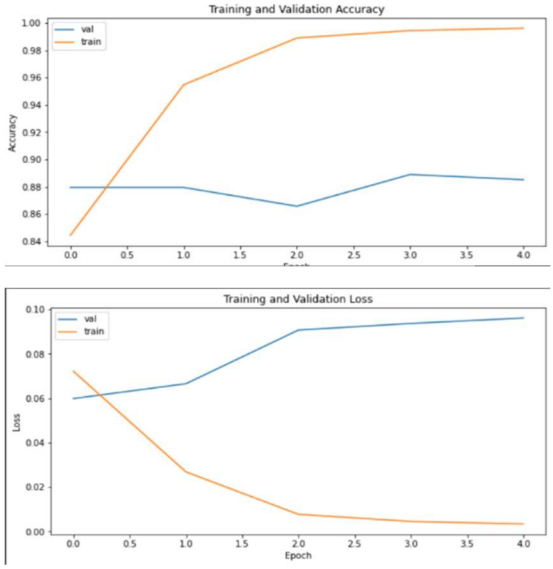

Fig. 7 Accuracy and loss graphic fine-tuned IndoBERTweet model when using dataset Ibrohim & Budi [11]

Fig. 7 shows the result of training and evaluating the model. Unlike Fig. 3 to Fig. 6, where the blue line represents training performance, in Fig. 7 and Fig. 8, the blue line represents evaluation performance, and the orange line represents training performance. The fine-tuned IndoBERTweet also suffers the same problem when training and evaluating using the Ibrohim & Budi dataset. In the first epoch, the evaluating model scores higher than in training. However, this score did not increase and instead decreased at epoch three and then increased at the next epoch and decreased again at last. Loss calculations increase when evaluating the model but decrease when training the model.

The result of training and evaluating the model with another data set is shown in Fig. 8. The model accuracy in evaluation is almost the same as when the model evaluates in the previous dataset. The pattern is the same as in Fig. 7, where the evaluation performance is higher at the first epoch than the training performance. At the next epoch, it increased but then decreased at epoch 3, then increased again until the last epoch. For the loss calculation, in evaluation, the model loss decreased not as much as when training but increased a bit at the last epoch.

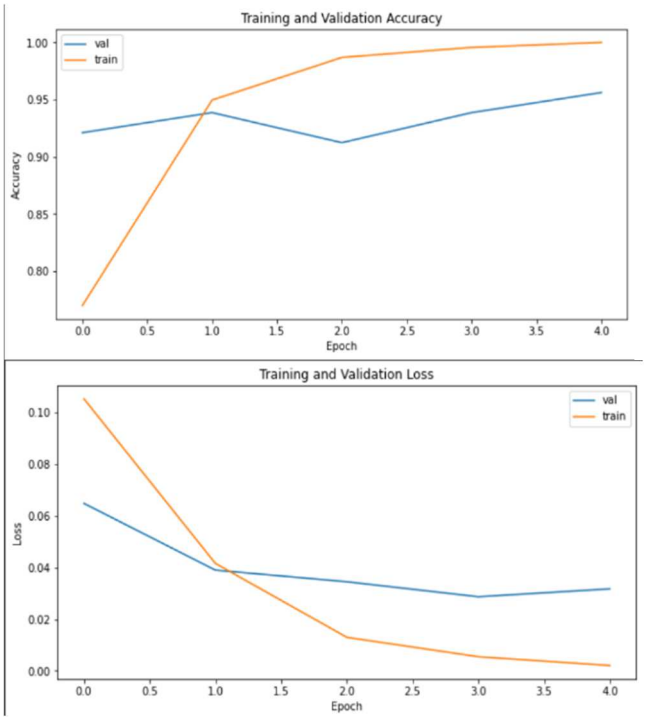

Fig. 8 Accuracy and loss graphic fine-tuned IndoBERTweet model when using dataset Alfina et al. [10]

Based on Table III to Table VIII, the calculation of precision, recall, accuracy, and F1-score is shown in Table IX alongside a comparison to another research. Overall, the model trained using the second dataset performed better than the first dataset. This is because in the second dataset, the type of language has been changed to a more formal word, making the sentence more proper in form. While in the first dataset, although it is provided with the vocabulary to convert informal to formal words, the data still had many informal words that affected the model training process. Our proposed IndoBERTweet + BiLSTM model achieved the best performance for both datasets.

TABLE IX
MODEL COMPARISON

Datasets	Model	Accuracy	Recall	Precision	F1-score
Ibrohim & Budi [11]	Fine-tuned IndoBERTweet	88.5%	88.3%	88%	88.1%
	IndoBERTweet + BiLSTM	**88.6%**	**88.5%**	**88.1%**	**88.3%**
	IndoBERTweet + CNN	88%	87.7%	87.5%	87.6%
	IndoBERTweet [9]	87.5%	-	-	-
	IndoBERT + BiGRU [21]	84.77%	-	-	-
Alfina *et al*. [10]	Fine-tuned IndoBERTweet	91.6%	91.5%	91%	91.2%
	IndoBERTweet + BiLSTM	**93.7%**	**92.9%**	**93.8%**	**93.3%**
	IndoBERTweet + CNN	90.2%	90%	89.5%	89.7%
	IndoBERTweet [9]	88.8%	-	-	-

This could happen because the IndoBERTweet model was trained using Twitter data by crawling Indonesian tweets using the official Twitter API.

Therefore, the BiLSTM layer helps the model to perform better. This was not the case with the CNN layer, as in our research, adding the CNN layer only improved the performance a bit, and the fine-tuned model still performed better than adding the CNN layer. Using the first dataset, our combined model with the BilSTM layer improved the performance slightly more than the fine-tuned IndoBERTweet. However, the second dataset significantly improved by 6% from the IndoBERTweet model and 5% increased performance from fine-tuned IndoBERTweet model.

When training our model, we used a small number of the epoch because the size of our dataset is relatively small and imbalanced. This small number of epochs helps the model not overfit; additionally, we add a dropout layer to reduce the overfitting. We also tried using epoch with large numbers, such as 10, 15, and 20. As a result, our model training accuracy keeps improving while validation accuracy could not keep up with training accuracy. Most of the time, the validation accuracy went up and down until our model finished training.

IV. Conclusion

The combined model has been able to classify hate speech properly through text processing, such as cleaning and converting slang and misspelled words to their original form and tokenizing the sentence. The token gained from tokenizer vocabulary using BertTokenizer and fed into the model. The result obtained from the model is evaluated using a confusion matrix. Based on the analysis, the model composed of IndoBERTweet and BiLSTM obtained much better results with the highest score of accuracy, recall, precision, and F1 score are 93.7%, 92.9%, 93.8%, and 93.3%, respectively. The model with IndoBERTweet and CNN, meanwhile, gained the lowest result with accuracy, recall, precision, and F1 scores of 88%, 87.7%, 87.5%, and 87.6%, respectively. Although the model used the CNN layer, there is no significant increase in performance gained from only using IndoBERTweet. There is also a slight difference between the results using different datasets. The difference varies between 2% to 5%, and the data quality, such as the type of word and the cleaning process, played a decisive factor in this matter. It is hoped that this research could lead to more research that focuses on datasets. Another use of other RNN layer is also a consideration to increase the performance of hate speech classification further.

References

[1] S. Kemp, "Digital 2021: Global Overview Report," 27 January 2021. [Online]. Available: https://datareportal.com/reports/digital-2021-global-overview-report. [Accessed 25 Maret 2021].

[2] W. Warner and J. Hirschberg, "Detecting Hate Speech on the World Wide Web," *Proceedings of the second workshop on language in social media,* pp. 19-26, 2012.

[3] J. W. Howard, "Free speech and hate speech," *Annual Review of Political Science,* pp. 93-109, 2019.

[4] B. Mathew, R. Dutt, P. Goyal and A. Mukherjee, "Spread of Hate Speech in Online Social Media," *Proceedings of the 10th ACM Conference on Web Science - WebSci '19,* pp. 173-182, 2019.

[5] H. I. Harahap, "Hate Speech in Election: Increasing Trends and Concerns," *Advances in Social Science, Education and Humanities Research,* pp. 44-46, 2019.

[6] A. Purnomo, "Legal perspectives concerning hate speech in indonesia," *PalArch's Journal of Archaeology of Egypt/Egyptology,* pp. 544-554, 2020.

[7] M. Mozafari, R. Farahbakhsh and N. Crespi, "A BERT-Based Transfer Learning Approach for Hate Speech Detection in Online Social Media," *COMPLEX NETWORKS 2019, SCI 881,* pp. 928-940, 2020.

[8] G. B. Herwanto, A. M. Ningtyas, I. G. Mujiyatna, I. N. P. Trisna and K. E. Nugraha, "Hate Speech Detection in Indonesian Twitter using Contextual Embedding Approach," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems),* pp. 177-188, 2021.

[9] F. Koto, J. H. Lau and T. Baldwin, "INDOBERTWEET: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization," *In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021),* pp. 1-9, 2021.

[10] I. Alfina, R. Mulia, M. I. Fanany and Y. Ekanata, "Hate Speech Detection in the Indonesian Language: A Dataset and Preliminary Study," *Proceeding of 9th International Conference on Advanced Computer Science and Information Systems 2017(ICACSIS 2017),* pp. 233-238, 2017.

[11] M. O. Ibrohim and I. Budi, "Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter," *Proceedings of the Third Workshop on Abusive Language Online,* pp. 46-57, 2019.

[12] F. D. Vigna, A. Cimino, F. Dell'Orletta, M. Petrocchi and M. Tesconi, "Hate me, hate me not: Hate speech detection on Facebook," *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17),* pp. 86-95, 2017.

[13] B. Gambäck and U. K. Sikdar, "Using Convolutional Neural Networks to Classify Hate-Speech," *Proceedings of the First Workshop on Abusive Language Online,* pp. 85-90, 2017.

[14] Z. Al-Makhadmeh and A. Tolba, "Automatic hate speech detection using killer natural language processing optimizing ensemble deep learning approach," *Computing,* pp. 501-522, 2019.

[15] H. Faris, I. Aljarah, M. Habib and P. A. Castillo, "Hate Speech Detection using Word Embedding and Deep Learning in the Arabic Language Context," *Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2020),* pp. 453-460, 2020.

[16] P. Kapil, A. Ekbal and D. Das, "Investigating Deep Learning Approaches for Hate Speech Detection in Social Media," unpublished, 2020.

[17] T. B. Nguyen, Q. M. Nguyen, T. H. Nguyen, N. P. Pham, T. L. Nguyen and Q. T. Do, "VAIS Hate Speech Detection System: A Deep Learning based Approach for System Combination," unpublished, 2019.

[18] S. S. Aluru, B. Mathew, P. Saha and A. Mukherjee, "Deep Learning Models for Multilingual Hate Speech Detection," unpublished, 2020.

[19] T. L. Sutejo and D. P. Lestari, "Indonesia Hate Speech Detection using Deep Learning," *International Conference on Asian Language Processing (IALP),* pp. 39-43, 2018.

[20] A. R. Isnain, A. Sihabuddin and Y. Suyanto, "Bidirectional Long Short Term Memory Method and Word2vec Extraction Approach for Hate Speech Detection," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems),* pp. 169-178, 2020.

[21] A. Marpaung, R. Rismala and H. Nurrahmi, "Hate Speech Detection in Indonesia Twitter Texts using Bidirectional Gated Recurrent Unit," *International Conference on Knowledge and Smart Technology (KST),* no. 13, pp. 186-190, 2021.

[22] R. Aggarwal, "Bi-LSTM," 4 July 2019. [Online]. Available: https://medium.com/@raghavaggarwal0089/bi-lstm-bc3d68da8bd0. [Accessed 12 August 2021].

[23] V. Kotu and B. Deshpande, Data Science (Second Edition), Elsevier, 2019.