Classifying Natural Disaster Tweet using a Convolutional Neural Network and BERT Embedding

1st Lucas Satria Aji Dharma
Dept. of Computer Science and Electronics
Universitas Gadjah Mada
Yogyakarta, Indonesia
lucasaji99@mail.ugm.ac.id

2nd Edi Winarko
Dept. of Computer Science and Electronics
Universitas Gadjah Mada
Yogyakarta, Indonesia
ewinarko@ugm.ac.id

***Abstract*—Social media platforms have become a medium to find a vast source of information throughout the internet. Twitter has become one of the more popular microblogging platforms out there, and the more users there are in these platforms means the more various types of information can be sent out in a day. On Twitter users are able to write their expression in the form of tweets, this will then create a post on twitter's timeline and other users are able to see these tweets. If a tweet suddenly gets viral, Twitter will put the user's tweets into the trending page allowing even more users to view the said tweet. During an event of a natural disaster often a lot of the tweets that are being posted, have mention of the disaster making it a trending topic on Twitter. From this, a vast number of tweets about a disaster can be collected as data, but not always are the tweets containing information about the disaster. Often there are tweets that use natural disaster words but do not actually talk about the disaster itself, hence are not informative and can be classified as a non-disaster tweet. This research paper aims to propose a system to classify the disaster tweets and the non-disaster tweet during a disaster. The proposed method is based on Convolutional Neural Network (CNN), using a Bidirectional Encoder Representation from Transformers (BERT) as an Embedding. As a comparison, it will then be compared with another embedding method named Word2Vec. The Evaluation result after training and testing of the CNN with BERT embeddings gave the most consistent results attaining accuracy of 97.16% precision of 97.63%, a recall of 96.64, and an f1-score of 97.13% for the model classification.**

Keywords*—*Deep Learning, NLP, Text Classification, Convolutional Neural Network, BERT

I. INTRODUCTION

Using twitter tweets as data has become one of the most important tools for Natural Language Processing (NLP) tasks. Especially with Twitter being used as a medium of communication every day, the amount of various information about different events that can be found is overwhelming [1]. Even with the overwhelming amount of data that can be found around twitter, often it is not enough just to use keywords alone to get useful tweets [2]. When only relying on a certain keyword we can get unnecessary information that may not even be related to the information that we are trying to gather. Therefore, a further breakdown and exploration of the tweet are needed to classify which of the tweets may contain useful information.

Disaster tweet classification study can be considered as a Natural Language Processing (NLP) task. The use of a deep learning model has started to become more common for natural language processing (NLP) tasks. Deep learning models such as a Convolutional Neural Network (CNN) have been proven to be a better classifier than a machine learning model such as Support Vector Machine (SVM) for handling text classification tasks [2], [3]. Working with a text classification task, the data that has been collected may be unstructured and therefore it undergoes a pre-processing task [4]. Pre-processing turns the unstructured data into much more structured and consistent data, this allows the data to be presented in a representational model [5]. Much of the previous research that has been conducted follows using a method such as a word embedding techniques include Continuous Bag of Words (CBOW) and Skip-gram model [6]. For the case of a disaster, this can be different, especially in a Twitter tweet the context of the sentence may be a factor for classification. With the approaches of word embedding to have been improved, a new method of text representation called contextual word embedding has arisen for the first time [7]. Here the authors introduced Bidirectional Encoder Representation from Transformers (BERT). What differentiates contextual word embedding from traditional word embedding is that contextual word embedding, can capture the context of the whole text, which in return can give better results than traditional word embedding [8],[9].

In recent studies revolving around the topic around the classification of natural disasters, most of them are conducted in the English language, such as a study that classified the informativeness of the disaster tweet which are then labeled into two classes: informative and non-informative while using deep learning method and word embedding [6]. Whereas classification in the Indonesian language has only been using the traditional Machine Learning method, at the time of this research being conducted. In one study, we have a dataset that focuses on detecting earthquake disasters using traditional machine learning methods [10]. This research study will focus on using the Indonesian language for the classification of natural disaster tweets while also implementing a deep learning model, Convolutional Neural Network (CNN), and using a Bidirectional Encoder Representations from Transformers (BERT) embedding, which will then be compared using Word2Vec embedding.

II. LITERATURE REVIEW

Twitter has become a widely popular platform for the collection of various types of information and using it as a data source for different purposes. The previous research focused on automatically identifying informative disaster-related tweets and filtering out uninformative ones [3]. They did this by using a Convolutional Neural Network (CNN) model to which the tweets are then marked as either informative or not informative. The CNN model is then compared with two supervised learning algorithms which are SVM and ANN. The study for this research is to improve the overall accuracy of which a deep learning model (CNN) is able to classify informative tweets during a disaster. In another paper, research was conducted focusing on the use of Twitter data to detect earthquakes [10]. They did this by using supervised learning which includes: Support Vector Machine (SVM), Random Forest, and Decision Tree. The dataset is based on Twitter feeds in the Indonesian language that is related to earthquakes. Several pre-processing techniques are also used such as tokenization, stop word removal, and normalization. The paper uses the recall evaluation to determine which of the proposed models will be better at detecting earthquakes. From the result, Random Forest was able to outperform SVM and Decision Tree.

Another research conducted focused on tweet data to classify informative tweets in disasters [6]. The aim of the research was to be able to differentiate tweets that might be beneficial and provide details about the people affected, disaster location, etc., and tweets that are not useful for a disaster. They proposed using a CNN and Word2Vec model to classify the tweets. The deep learning model is shown to be prominent at classification tasks especially with the use of word embeddings, in research conducted by Madichetty et al. [1] conducted another much more recent research that uses ELMo embedding, a type of contextual representation for the classification of disaster-related tweets. The proposed ELMo embedding is combined with two dense classifiers which are the Rectifier Linear Unit (ReLU) activation function and the SoftMax function. The performance of the said proposed model will then be compared with three baseline models which are: SVM with Bag-of-words (BoW), CNN with crisis word embedding, and MLP-CNN.

III. METHOD

From reviewing previous studies and research, the proposed model of this research will aim to use a deep learning model for classifying natural disaster tweets by using a Convolutional Neural Network (CNN) while also using Bidirectional Encoder Representation from Transformers (BERT) as the word embedding to classify the tweets information as the output. The proposed model hopes to improve the performance of a text classification task. The following section will describe the approaches of the dataset that is used, followed by the deep learning model.

A. *Dataset*

The natural disaster dataset will be based on Twitter tweets in the Indonesian language, the collection of tweets will use various natural disaster-related keywords in the Indonesian language which are then categorized as Earthquake, Flood, Pyroclastic flow, Drought, Typhoon, Tsunami, Landslide, Eruption. Different natural disaster keywords are used to ensure the diversity of the type natural disaster model and not locked into one specific type of natural disaster. The total dataset that is collected contains 6049 tweets, to which 3038 of those are labeled as disaster and 3011 is labeled as a non-disaster type. Table 1 shows an example of the data labeling within the dataset.

TABLE I. SAMPLE OF THE LABELLED DATA

Tweet	Label
Earthquake Magnitude: 5.2, 15-Jun-21 21:16:14 WIB, Location: 4.33 South Latitude, 102.37 East Longitude (Earthquake epicenter was at sea 36 km Southwest of Seluma), Depth: 26 Km Felt (MMI) III-IV Seluma, II - III Kaur, III Bengkulu City, III Central Bengkulu, III Kepahiang, II Curup, II Liwa #BMKG	1
The period 00.0013.30 WIB does not occur hot clouds avalanches, please always follow the official info from our account.	0
BMKG stated that the boom that was heard in Jabodetabek was not from the earthquake #kumparannews	0
If the earthquake coincides with the corona, do we have to leave the house or stay indoors?	0
11.08 The forest fire incident in the northwest sector of Merapi this morning was not caused by hot clouds, but falling material (ballistics) which was still hot. People should stay calm, so far Merapi has not emitted hot clouds. #statusalert	1

The disaster tweet is then manually classified with a Boolean value where the value 1 will be classified as a disaster, and a non-disaster will be labeled as 0. A problem that arises when doing a manual classification is the perspective viewing the texts of the tweets. Hence, two annotators are asked to label the tweets based on their own perspective on how they view the disaster tweet. Each labeled dataset is then collected and is then discussed the relatively of the tweets. The agreed dataset to be used in this study involves tweets that involve an area of a disaster such as the whereabouts and the impact of the disaster will be classified as a disaster tweet. Non-disaster tweet involves tweets that are seen as conservative without the context of the impact of the disaster. Furthermore, the labeled dataset will be categorized based on the type of disaster that is present within the text, this can be seen in Table 2. Overall, there is a total of nine categories. The dataset will later be split for training, validation, and testing, where 60% will be used for training, 20% will be used for validation, and 20% for testing.

TABLE II. CATEGORIZED CLASSIFIED TEXTS

Texts	Category
The earthquake was felt, the magnitude of the depth was kilometers, February, West Indonesia, the coordinates of the epicenter of the sea kilometers southwest of Padang Sidempuan	Earthquake
Floods in Mojokerto Regency, Central Java Province, people are affected by material losses	Flood
Avalanche activity changes the recommendations for the potential danger of hot cloud lava avalanches in the southwest sector power covers the yellow river	Pyroclastic flow
Ash rain from the eruption of Mount Merapi is reported to have flooded the area of North Sleman Regency, BPBD	Eruption
Threatened by a large tsunami, Banten residents are considered unprepared	Tsunami
Drought increases peatland in Muaro Jambi burns	Drought
People die as landslides hit houses in India, Nepal	Landslide
The number of houses damaged by the tornado in Pasuruan has increased	Typhoon
Get better soon, Mother Earth, the highest peak, Mahameru, Merapi, the BMKG Krakatoa earthquake my Indonesia is Semeru	Others

B. Pre-processing

Raw data of a Twitter tweet can contain not just text but also images, emojis, gifs, URLs, etc. Since the focus of this research is on the classification of textual data, unwanted content needs to be removed. Hence, the dataset will need to be pre-processed, to make the text data uninformed, this means removing URL links, mentions, and emojis. In this study stemming and the removal of stop words will be used for comparing the dataset that did not undergo stop words removal and stemming. Thus, there will be two datasets where the first one will not be using stop words removal and stemming, while the second will be using stop words removal and stemming. Table 3 shows the pre-processing techniques that will be used.

TABLE III. EXAMPLE PROCESS OF THE PRE-PROCESSING TECHNIQUE

Pre-processing	Before text	After text
Case-folding	PART FLOOD recedes in SOUTH SULAWESI	Some floods recede in South Sulawesi
Removal special symbols	@bmkg was there an earthquake in the south of Jogja. #staysafe #shockedearthquake	BMKG, was there an earthquake in the southern area of Jogja, #stay safe #shockedearthquake
Tokenization	Flooding again like what is happening in Kalimantan right now	flood, again, like, which, now, is happening, in, Kalimantan, right
Stop words removal	Tornadoes damaged 8 houses and 2 schools in Blitar	Tornado destroys Blitar school house
Stemming	Hundreds of houses collapsed, hit by a tornado	hundreds of houses collapsed in a tornado

C. Bidirectional Encoder Representation from Transformers (BERT) Embedding

The new model introduced for language representation is Bidirectional Encoder Representations from Transformers (BERT) [7]. As stated from the name it utilizes a transformer to create the vector representation, allowing it to scan an entire sequence and predict the masked word. This research will be using a pre-trained Indonesian language BERT model, namely IndoBERT [11]. By using IndoBERT it will be able to process the Indonesian language on which the dataset will be based, this will then allow BERT to be able to predict the context of the words within the text of the sentence. IndoBERT does this from the two unsupervised learning: Masked LM and Next Sentence Prediction (NSP) during the pre-training. Attaining the BERT embedding first the text dataset needs to be tokenized from the pre-trained BERT tokenizer. Tokenizing the text will allow BERT to add the special CLS token at the beginning of the text. The tokenized texts can then be fed into the pre-trained BERT model, the outputs are then the feature vectors that will be used as the embedding. The embeddings are then sent into the convolutional layer for classification. Figure 1 shows the proposed architecture of the BERT embedding used for the CNN model.

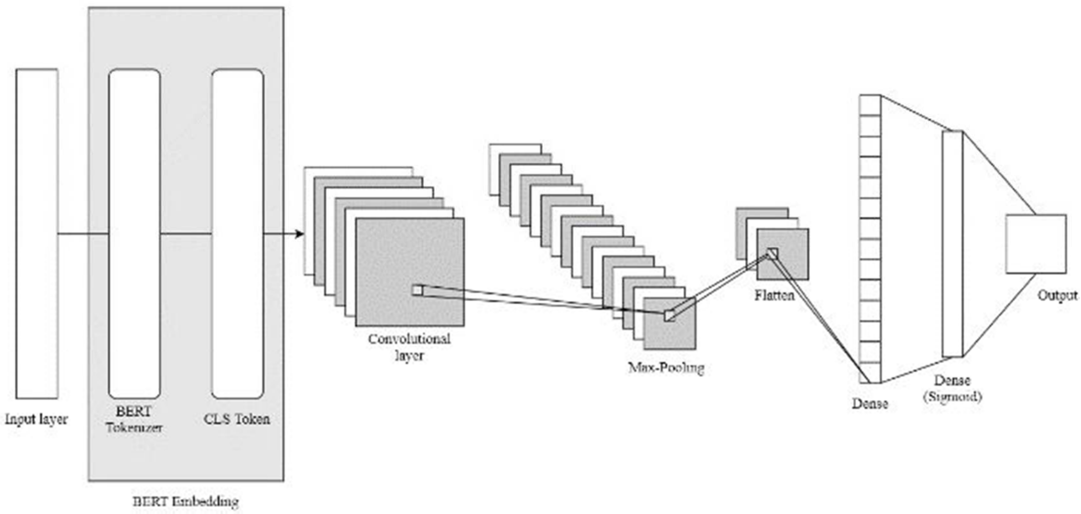

Fig. 1. CNN model architecture using BERT embedding

D. Convolutional Neural Network

CNN was initially a deep learning technique for visualizing images in computer vision, but it has recently been used more in the work of NLP and can be used to perform text classification [12]. CNN can represent the documents as a row of matrices with the vector representational in this case it's the pre-trained word embedding thus, allowing it to be learned efficiently.

For the use of text classification, CNN consists of a 1-dimensional convolutional layer followed by the max-pooling layer objectively tries to reduce the feature of the spatial size within the convolution without having to lose important values of the feature map [13]. Taken into account, by means of its convolution and max-pooling layer, CNN can capture the most important n-gram data while also retaining the important aspect of the feature map. The most accurate weight can also be determined through the fully-connected layer where it undergoes a backpropagation process and each neuron is prioritized by the given weight of the label.

In this study, the proposed CNN model will have its hyperparameters tuned to get the most optimized results. For this research, the CNN model will consist of the following layers while using different types of parameters until reaching the most optimal hyperparameter using the Keras tuner which are:

- Input layer: This consists of the pre-processed dataset that will be used for the classification.
- Embedding layer: Here the proposed pre-trained embedding layer will consist of the BERT feature vectors.
- Convolution layer: The size of filters will have an option of using 64, 128, and 256 filters, using a kernel size of either 2, 3, and 4.
- Max Pooling layer: Takes the maximum number of the n-dimensional matrix output using a size of either 2, 3, or 4.
- Dropout layer: Preventing overfitting within the model with optional sizes of 0.2, 0.3, and 0.4.
- Dense layer: We Will be using two dense layers which will become the fully connected layer that will be used to classify the features and calculate the sigmoid. The first dense layer will have an option unit of either 64, 128, or 256, while also using an L2 regularizer with the default size of 0.01. The final dense layer will have a unit of 1 and use the Sigmoid activation function since it is a binary classification.

The proposed architecture for the CNN model can be seen in Figure 2.

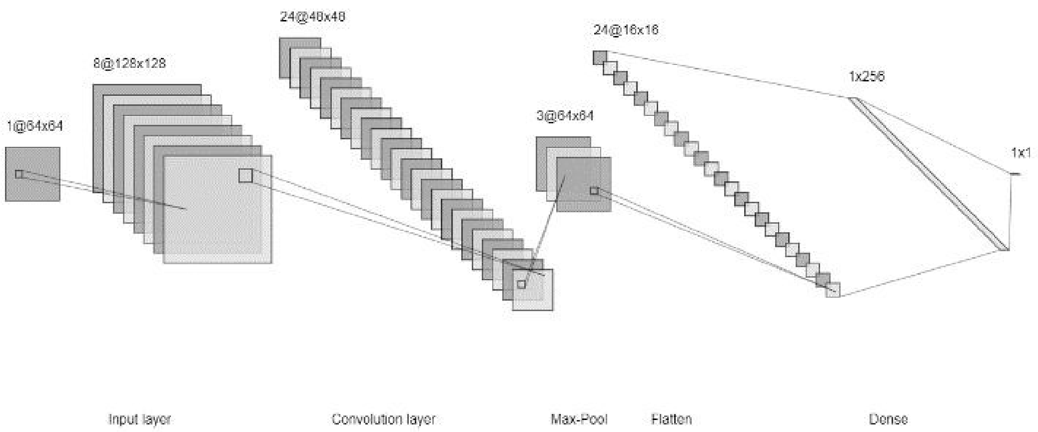

Fig. 2. Proposed CNN model architecture created using LetNet style

E. Keras-tuner

The Keras library offers a Keras-tuner which allows us to tune the model hyperparameters to be able to give us the most optimal hyperparameters. Using Keras-tuner allows us to give a range of sets to the parameter in our model such as setting the number of filters to use in the Conv layer. There are three types of hyper tuning methods that are often used which are Random Search, Hyperband, and Bayesian Optimization. But in this study, it is using Bayesian Optimization due to it being able to compensate for the problem in Random Search. Unlike Random Search where it keeps choosing random values for each iteration Bayesian only uses Random values during the first few trials, which after then will record every previously trained model at which it will use the results to adjust and change the next possible hyperparameters for better results than the previous. The Keras-tuner is put on trial 10 times to ensure that the most optimal hyperparameters are achieved, the optimal hyperparameter will then be used to build the CNN model for training and testing.

IV. Result And Discussion

The implementation of this research is using the Python language to create the deep learning model. The proposed model in this study has been trained and tested using the two datasets of different pre-processing techniques. An evaluation of the performance evaluation and its result will be done and will be described in this section.

A. Data Pre-processing and Exploration

Once the dataset has undergone pre-processing, there was a slight decrease of value in data from 6049 to 5978 tweets, this was due to the removal of duplicates. The results from the data labeling are seen to be fairly balanced from the two classes, as it is known there are two labeled classes of a Boolean integer where 1 is classified as disaster and 0 is a non-disaster. As seen from Figure 3 a total of 2970 is labeled as a disaster (1), hence giving the remaining non-disaster (0) label to 3008. Here, the data is slightly unbalanced where the non-disaster label is greater than the disaster label.

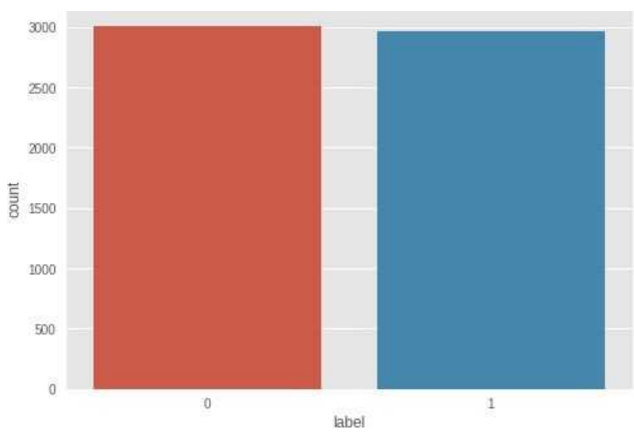

Fig. 3. Distribution of data label

The tweets that are labeled as a disaster are shown to be detailed since they are collected from one of the Indonesian Meteorology, Climatology, and Geophysical Agencies. Hence, details such as the whereabouts of the location and how powerful the disaster is written in the tweets. This can be seen from the frequent occurrence of the keyword that can be seen in Figure 3. The word cloud displays that earthquake natural disaster is the most common disaster word within the dataset; this could be the one reason why the other common words are related to describing an Earthquake tweet.

Fig. 4. Word cloud of the most frequent keywords within the dataset

B. Performance analysis

Implementation of the CNN model is tested with the two datasets, the first dataset that will be used will undergo stop words removal and stemming while the second dataset will not. The main model CNN with BERT embedding will be compared with Word2Vec embedding.

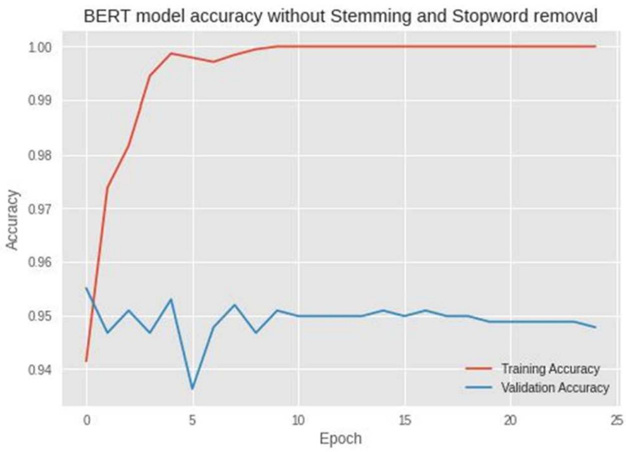

Fig. 5. Model accuracy graph during training and validation using dataset without stemming and stop words removal

As seen Figure 5 it shows model accuracy of the BERT embedding using a dataset that did not undergo stemming and stop words removal during training and validation. It can be seen that during training the model is able to reach a high training of 99.95% while remaining consistent to almost reaching the highest accuracy, while with the validation accuracy maintains a steady result of around 95%, but shows some fluctuation and a major decrease during its 5th epoch. On the other hand, Figure 6 shows the model loss using a model it can be seen that the model has good training stability until around the 6th epoch when the validation loss starts to increase from the graph which shows that model is overfitting and thus the model could have been stopped at an earlier epoch.

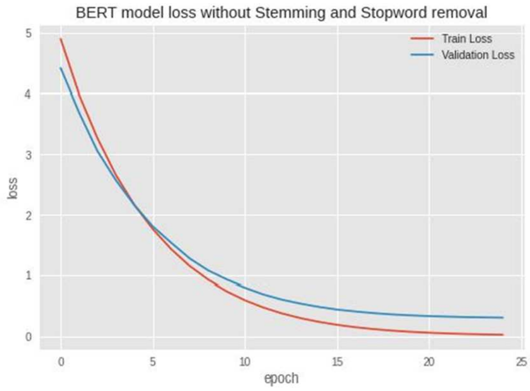

Fig. 6. Model loss graph during training and validation using dataset without stemming and stop words removal

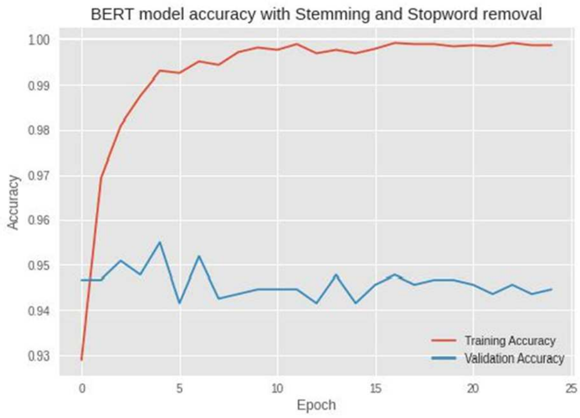

Fig. 7. Model accuracy graph during training and validation using a dataset with stemming and stop words removal

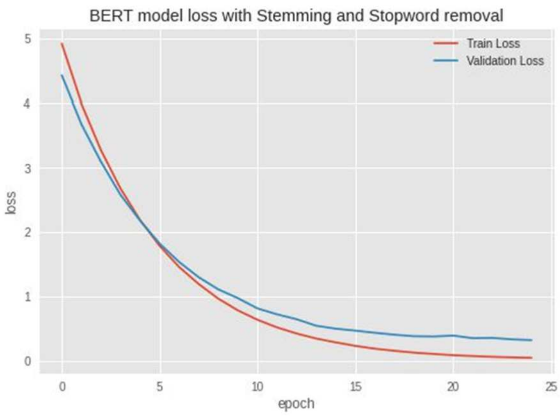

Fig. 8. Model loss graph during training and validation using a dataset with stemming and stop words removal

Comparing the BERT embedding model using the dataset that is using stemming and stop words removal, it can be seen in Figure 7 the model accuracy using the said dataset. Depicting it from graph it can be seen that it shows similarity during training and validation, where training accuracy is able to reach nearly 99.00% while achieving a validation accuracy of 94.67%, slightly lower than using the previous dataset. While its model loss graph from Figure 8 shows similar result from the previous model where it also begins to overfit at an early epoch.

A classification report can be created to calculate the Precision, Recall, and F1-Score. Both results for the classification report are tested using the same parameter for the CNN and BERT embedding model, Table 4 shows the comparison of the model using the dataset that did not undergo stemming and stop words removal.

TABLE IV. MODEL EVALUATION WITHOUT STEMMING AND STOP WORDS REMOVAL

Embedding	Accuracy	Precision	Recall	F1-Score
BERT	**97.16**	**97.63**	**96.64**	**97.13**
Skip-gram	95.65	97.39	93.79	95.56
CBOW	94.90	96.36	93.29	94.80

From Table 4, it can be seen that BERT almost excels in all of the evaluation scores, while Skip-gram is just slightly behind when it comes to the result of its recall. While CBOW is slightly behind in performance compared to BERT and Skip-gram. To compare this result, the second dataset which was stemmed and its stop words removed can be seen in Table 5, it shows the results of the model using the dataset that has undergone stemming and stops words removal.

TABLE V. MODEL EVALUATION WITH STEMMING AND STOP WORDS REMOVAL

Embedding	Accuracy	Precision	Recall	F1-Score
BERT	96.07	96.29	95.80	96.05
Skip-gram	95.23	98.21	92.11	95.06
CBOW	95.15	98.38	91.78	94.97

The results of training and testing of the CNN model using BERT embeddings and Word2Vec embeddings showed that these two types of embedding show different performances under certain different conditions. During the training phase, BERT can achieve the highest training accuracy while using the dataset that did not undergo stemming and stop words removal during preprocessing. While also achieving a constant validation accuracy of around 95% using the same dataset, the BERT model did show slight overfitting after its 5th epoch and as a result of this, it causes fluctuation to the testing accuracy which can be observed also during the 5th and 12th epoch.

In the case of BERT, it was able to perform better and give better results using the dataset that was not stemmed and having its stop words removed. While using Skip-gram it did not matter which dataset to use, as there is very little in changes of performance when tested using the two datasets. CBOW on the other hand was able to perform better using the dataset that was stemmed and had its stop words removed.

TABLE VI. CONFUSION MATRIX OF CNN WITH BERT EMBEDDING

	Pred. Non-disaster	Pred. Disaster
True Non-disaster	586	14
True Disaster	17	579

TABLE VII. CONFUSION MATRIX OF SKIP-GRAM WORD2VEC

	Pred. Non-disaster	Pred. Disaster
True Non-disaster	593	7
True Disaster	37	559

TABLE VIII. CONFUSION MATRIX OF CBOW WORD2VEC

	Pred. Non-disaster	Pred. Disaster
True Non-disaster	589	11
True Disaster	37	559

Evaluating from the confusion matrix BERT is better at predicting the disaster whereas Word2Vec can predict better for a non-disaster, as seen in table the SG predicted 37 of them being a non-disaster when it is supposed to be a disaster label. Similarly with BERT it shows more error when predicting the non-disaster label.

Fig. 9. Confusion matrix of the categorized texts for CNN with BERT Embedding

TABLE IX. TEXT FROM GEMPA, BANJIR, AND OTHER CLASSES AS FALSE POSITIVE BY THE CNN MODEL

Actual class	Predicted class	Text
Other	Earthquake	Indonesia is different, it's an earthquake but not the effect of the 'child of Krakatoa' just check
Other	Earthquake	The story of Rasulullah SAW which has something to do with mountains I think is right with the current phenomenon, the booming earthquake of Anak Krakatau, Mount Krakatoa
Landslide	Flood	heavy rains caused flooding as many as five units of residents' houses in the village of Babakan Cingkeuk, the RT of the pillars of the residents of Cibokor Village, Cibeber District, Cianjur Regency were damaged heavy from being hit by a landslide
Landslide	Flood	very heavy rains caused floods & landslides in the area Padang City, West Sumatra Province

Here, if we look at the confusion matrix based on the category of the text, it can be seen the earthquake the class has the most classification as it was the most common label among the other classes, which also contains several errors where it is classified as a different class than its original, this can be called the False Positive. This can be seen where 28 'other' classes are predicted as earthquakes. The model should have predicted it as a negative value (other) but instead has predicted it as a positive (earthquake) hence getting a false positive. Another case of a false positive is happening in the flood class where 16 of the texts are predicted as landslides. A few examples of the false 'positives' cases from earthquake, flood, and other classes can be seen in Figure 9.

Viewing the other classes, the tsunami class has encountered a False Negative result; this means that the model predicted the value as negative (tsunami) instead of being the positive (tsunami). Hence, giving the wrong predictions which can be seen from the "tsunami" class where 18 of them are predicted as earthquake class. In the case of the tsunami class being classified as an earthquake, class is due to how a tsunami often occurred, since usually during a disaster earthquake it can be followed by a tsunami if the earthquake happens in the sea. Hence, the model predicted it as an earthquake since there is a very close similarity especially when there is a text announcing the event of an earthquake it is often followed by a tsunami. To further view how the model can predict such classes it can be seen in Table 10 where it shows an example of the false-negative from the tsunami as predicted as earthquake.

TABLE X. FALSE-NEGATIVE TEXT FROM TSUNAMI CLASS PREDICTED AS EARTHQUAKE CLASS

Actual class	Predicted class	Text
tsunami	earthquake	An earthquake with a magnitude of November, western Indonesia time, LS BT, kilometers southwest of the Binuangeun estuary, Banten, a depth of kilometers, no tsunami potential BMKG
tsunami	earthquake	An earthquake with a magnitude of 1,200 kilometers, October, western Indonesian time, coordinates LS BT, kilometers northwest of Poso, Central Sulawesi, no tsunami potential BMKG
tsunami	earthquake	earthquake magnitude august western Indonesia time location ls bt kilometer northeast of Banggai islands central Sulawesi depth kilometers no tsunami potential BMKG

In the case of the tsunami class being classified as an earthquake, class is due to how a tsunami often occurred, since usually during a disaster earthquake it can be followed by a tsunami if the earthquake happens in the sea. Hence, the model predicted it as an earthquake since there is a very close similarity especially when there is a text announcing the event of an earthquake it is often followed by a tsunami.

V. CONCLUSION

CNN with pre-trained BERT embedding was able to get the best result when using a dataset that did not have its stop words removed and applying stemming. Being able to achieve the most consistent result of the accuracy, precision, recall, and f1-score. As for Skip-gram, the dataset did not matter as it was able to achieve similar results using the two datasets and able to outperform the BERT embedding and CBOW using the dataset that was stemmed and had its stop words removed. Proven by the evaluation of this research the use of a stop words removal can decrease the overall performance of the model. This is since stop words removal reduces the size of the data set and for a text, this often can change the overall meaning within the text, even though it reduces the training time. Therefore, the use of stop words removal is often not necessary and having more data set size is better for the model as it can improve the overall performance of the model. Pre-trained BERT without fine-tuning for downstream tasks may perform similarly with Word2Vec trained on downstream tasks. As seen from the result of the research both types of embedding gave similar results. The preprocessing technique that is being used plays an important role when it comes to Deep Learning classification as it would later affect the model's ability to classify the dataset during training and testing.

Since this study did not involve fine-tuning the BERT model, for future works fine-tuning should be taken into consideration as this could affect changes in the result within the model. Another approach that can be done for this research is by using another type of artificial neural network architecture besides CNN and can be implemented to be used as a comparison, while also using the same dataset. The dataset for this study can be further labeled as a multi-class classification as there are different types of natural disasters, and instead can predict or classify the type of natural disaster.

REFERENCES

[1] S. Madichetty and M. Sridevi, "Improved classification of crisis-related data on Twitter using contextual representations," Procedia Comput. Sci., vol. 167, pp. 962–968, 2020.

[2] D. T. Nguyen, K. A. Al Mannai, S. Joty, H. Sajjad, M. Imran, and P. Mitra, "Rapid classification of crisis-related data on social networks using convolutional neural networks," arXiv Prepr. arXiv1608.03902, 2016.

[3] C. Caragea, A. Silvescu, and A. H. Tapia, "Identifying informative messages in disaster events using convolutional neural networks," in International conference on information systems for crisis response and management, 2016, pp. 137–147.

[4] A. Hotho, A. Nürnberger, and G. Paass, "A brief survey of text mining.," in Ldv Forum, 2005, vol. 20, no. 1, pp. 19–62.

[5] B. S. Harish, D. S. Guru, and S. Manjunath, "Representation and classification of text documents: A brief review," IJCA, Spec. Issue RTIPPR, pp. 110–119, 2010.

[6] P. Bhere, A. Upadhyay, K. Chaudhari, and T. Ghorpade, "Classifying Informatory Tweets during Disaster Using Deep Learning," in ITM Web of Conferences, 2020, vol. 32, p. 3025.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv Prepr. arXiv1810.04805, 2018.

[8] A. Romascanu et al., "Using deep learning and social network analysis to understand and manage extreme flooding," J. Contingencies Cris. Manag., vol. 28, no. 3, pp. 251–261, 2020.

[9] M. S. Jagadeesh and P. J. A. Alphonse, "NIT_COVID-19 at WNUT-2020 Task 2: Deep Learning Model RoBERTa for Identify Informative COVID-19 English Tweets.," in W-NUT@ EMNLP, 2020, pp. 450–454.

[10] P. Rendra Dwi Lingga, Chastine Fatichah, & Diana Purwitasari. "Deteksi Gempa Berdasarkan Data Twitter Menggunakan Decision Tree, Random Forest, dan SVM". Jurnal Teknik ITS, 6, 153-158. 2017.

[11] B. Wilie et al., "Indonlu: Benchmark and resources for evaluating indonesian natural language understanding," arXiv Prepr. arXiv2009.05387, 2020.

[12] Yoon Kim. "Convolutional Neural Networks for Sentence Classification". In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1746-1751). 2014.

[13] Van Quan Nguyen, Tien Nguyen Anh, & Hyung-Jeong Yang. "Real-time event detection using recurrent neural network in social sensors". International Journal of Distributed Sensor Networks, 15, 1550147719856492. 2019.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," arXiv Prepr. arXiv1301.3781, 2013.

[15] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations, 2020, pp. 38–45

[16] ALRashdi, R., & O'Keefe, S. "Deep Learning and Word Embeddings for Tweet Classification for Crisis Response". Retrieved from https://arxiv.org/abs/1903.11024. 2019.

[17] M. Imran, S. Elbassuoni, C. Castillo, F. Diaz, and P. Meier, "Extracting information nuggets from disaster-Related messages in social media.," Iscram, vol. 201, no. 3, pp. 791–801, 2013.

[18] N. A. Smith, "Contextual word representations: A contextual introduction," arXiv Prepr. arXiv1902.06006, 2019.

[19] A. F. Hidayatullah and M. R. Ma'Arif, "Pre-processing tasks in Indonesian Twitter messages," in Journal of Physics: Conference Series, 2017, vol. 801, no. 1, p. 12072.

[20] B. E. Parilla-Ferrer, P. L. Fernandez, and J. T. Ballena, "Automatic classification of disaster-related tweets," in Proc. International Conference on innovative engineering technologies (ICIET), 2014, vol. 62.

[21] S. Madichetty and M. Sridevi, "Detecting informative tweets during disaster using deep neural networks," in 2019 11th International Conference on Communication Systems & Networks (COMSNETS), 2019, pp. 709–713.

[22] T. R. Luthfitama, "Pengenalan Entitas Bernama pada Tweet tentang Bencana Menggunakan Conditional Random Field." Universitas Gadjah Mada, 2019.