mathematics

MDPI

Article

A Comprehensive Analysis of Transformer-Deep Neural Network Models in Twitter Disaster Detection

Vimala Balakrishnan [1,*], Zhongliang Shi [1], Chuan Liang Law [2], Regine Lim [1], Lee Leng Teh [1], Yue Fan [1] and Jeyarani Periasamy [3]

[1] Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[2] Malayan Banking Berhad, Kuala Lumpur 50050, Malaysia
[3] Faculty of Data Science and Information Technology, INTI International University, Nilai 71800, Malaysia
* Correspondence: vimala.balakrishnan@um.edu.my

Abstract: Social media platforms such as Twitter are a vital source of information during major events, such as natural disasters. Studies attempting to automatically detect textual communications have mostly focused on machine learning and deep learning algorithms. Recent evidence shows improvement in disaster detection models with the use of contextual word embedding techniques (i.e., transformers) that take the context of a word into consideration, unlike the traditional context-free techniques; however, studies regarding this model are scant. To this end, this paper investigates a selection of ensemble learning models by merging transformers with deep neural network algorithms to assess their performance in detecting informative and non-informative disaster-related Twitter communications. A total of 7613 tweets were used to train and test the models. Results indicate that the ensemble models consistently yield good performance results, with F-score values ranging between 76% and 80%. Simpler transformer variants, such as ELECTRA and Talking-Heads Attention, yielded comparable and superior results compared to the computationally expensive BERT, with F-scores ranging from 80% to 84%, especially when merged with Bi-LSTM. Our findings show that the newer and simpler transformers can be used effectively, with less computational costs, in detecting disaster-related Twitter communications.

Keywords: disaster; Twitter; deep neural network; transformers; ensemble

MSC: 68-04

Citation: Balakrishnan, V.; Shi, Z.; Law, C.L.; Lim, R.; Teh, L.L.; Fan, Y.; Periasamy, J. A Comprehensive Analysis of Transformer-Deep Neural Network Models in Twitter Disaster Detection. *Mathematics* **2022**, *10*, 4664. https://doi.org/10.3390/math10244664

Academic Editors: Alvaro Figueira and Francesco Renna

Received: 18 July 2022
Accepted: 29 August 2022
Published: 9 December 2022

1. Introduction

Social media has become a common place for people to seek information and help during emergencies and major crises, particularly during natural disasters such as storms, tsunamis, earthquakes, flood, etc., by sharing posts in the forms of images, texts, and videos [1,2]. The platforms have a developing role in how people communicate and respond to disasters, providing a network to seek help; gain information, guidance and reassurance; and respond to help requests. For instance, social media posts requesting aid and support were found to have superseded the emergency (911) phone system during the 9/11 American "natural disaster" [3]. Similarly, "#SOSHarvey" and "#HelpHouston" were found to be trending during Hurricane Harvey and were used to flag people who needed help/rescue [3].

Twitter, in particular, has been shown to be popular in generating disaster-related content, with users tweeting information about affected people and infrastructure damage that are sometimes very useful for aid and rescue teams, government, and private disaster relief organizations rendering assistance to those in need. The social media platform is known for its ability to communicate quickly across space supporting victims and disaster response, directing resources, and highlighting what the affected community prioritizes in a disaster [2–6]. As a matter of fact, Twitter is regarded as the 'most useful social media

tool', particularly for natural disasters [2,6,7]. Despite its popularity, tweets are limited in terms of their length, and thus tend to be more challenging (e.g., sparse, more abbreviations, etc.), thus making it difficult to differentiate if a specific communication (i.e., tweet) is related to a disaster or not.

Social Media and Disaster Detection

Although the data generated by social media platforms such as Twitter are ubiquitous, extracting useful and relevant information is not only a tedious task, but nearly impossible due to their enormous volume and velocity, thus making automatic disaster detection and classification models feasible solutions [2,7–9]. With the advent of artificial intelligence (AI), evidence exists showing that approaches such as machine and deep learning can be effectively used to detect information related to natural disasters, based on social media communications [7]. A search of the literature on the application of AI and social media in disaster-related events revealed several aspects investigated by research scholars, including damage assessments [10,11], enhancing or promoting situational awareness [5,6], using sentiment for disaster predictions [8,12,13], and disaster classification/detection models focusing on differentiating informative and non-informative content [2,8,9,14], the latter of which is the focus of the present study. For example, the authors of [10] used a semi-supervised approach to evaluate the damage extent indicated by Twitter communication during the Hong Kong and Macau typhoons in 2017, whereas the authors of [12] used a big data driven approach for disaster response through sentiment analysis, with the results indicating a lexicon-based approach to be more suitable in analyzing the needs of the people during a disaster.

Numerous studies were found to have used the traditional machine learning algorithms, such as support vector machine (SVM), naïve Bayes, decision tree, and logistic regression, etc., with promising results [15–18]. More recent studies utilized deep learning algorithms such as the convolutional neural network (CNN) and the recurrent neural network (RNN) [19–21]; however, most of them were based on the context-free word embedding techniques, such as Word2Vec. In this technique, the context in which a word is used is disregarded. For example, *"#RockingBand, fire and smoke on stage, having a blast at this concert!!!"* indicates that the user is having a great time at a concert despite the use of words such as "fire," "smoke," and "blast." The tweet is not disaster-related; however, a context-free word embedding technique will likely classify the tweet as such, due to the occurrence of these words. To address this issue, some studies refined the models' performance using a contextual or transformer-based word embedding technique, that is, bidirectional encoder representations from transformers (BERT) [8,9,22–25]. However, studies exploring these transformers, including BERT variants such as RoBERTa, AlBERT, and ELECTRA, for example, are scant, despite the growing popularity of these contextual word embedding techniques and their positive results [22,26].

To close this gap and to further extend the current literature, we aim to explore several well-known deep learning algorithms, especially neural network (NN) models and transformer-based word embedding techniques, to identify the best/optimal ensemble solutions in detecting Twitter disaster-related communications. The main contributions of this study are as follows:

- We explore, implement, and compare the transformer-based embedding techniques, including the base model and its simpler variants, in detecting disaster using a real-life Twitter dataset;
- We implement the various transformers with several well-known NN models, and identify the best/optimal combination in detecting disasters via Twitter.

From the above-mentioned comparisons and analyses, the study offers evidence and support to alternative solutions, including the use of the simpler and more cost-effective transformer variants in effectively detecting disaster-related communication on social media.

The remainder of the paper is structured as follows: related studies employing deep learning approaches, using both context-free and contextual based techniques, are reviewed in Section 2, followed by the methods adopted to achieve the main objective of this study. Section 4 provides the results and a discussion, and the conclusion is provided in Section 5, along with limitations and ideas for future directions.

2. Related Work

Deep learning models are generally based on supervised models requiring large amounts of labeled data, often yielding more accurate results, albeit being computationally expensive [14,26]. Popular examples of deep NN models include CNN, RNN, and long short-term memory (LSTM), among others. A search of the literature revealed several studies exploring and proposing deep learning models to detect social media-based disaster identification. For example, Yu et al. [19] found their CNN model to yield the best F-score (i.e., 80%) based on experiments conducted using three datasets, namely, Hurricane Harvey, Hurricane Sandy, and Hurricane Irma. A similar study using CNN was done by the authors of [20], who extracted location references from emergency tweets, with findings indicating an F-score of 96%. The authors further extended their work by incorporating a multi-modal technique using LSTM and found the combination of both text and images to produce the best F-score (i.e., 93%) compared to using only text (i.e., 92%) [21]. Another study based on the CrisisLexT26 dataset containing tweets related to 26 disasters, trained and tested using Bi-directional Gated Recurrent Units (GRU) and LSTM learning models, reported that both the models detected disaster-related tweets, with F-score values of 79% and 82% for LSTM and Bi-GRU, respectively.

Others explored conventional word embedding techniques, such as the authors of [2], who proposed a hybrid CNN model combining character and word embedding techniques (i.e., FastText) to detect disaster tweets using datasets related to hurricanes, floods, and wildfires. They found that character-based CNN performed the best across all the datasets, with an average F-score of 71%. Conversely, the authors of [27] proposed a multilayer perceptron model, which is a feed forward NN using Word2Vec embedding, to classify disasters using two earthquake datasets for training. The authors tested their model using a COVID-19 dataset and reported a weighted F-score of 85%. Although often reported to yield improved detection results, these conventional word embedding techniques are context-free; hence, the word "fire" would be assumed to have the same meaning, regardless of its use in a sentence [9,26]. To address this issue, the contextual embedding learning model BERT was proposed by Devlin et al. [28].

BERT is a pretrained transformer bidirectional training model developed to resolve language modeling and next sentence prediction in tasks involving natural language processing (NLP) [28]. Unlike the conventional word embedding techniques, such as FastText, GloVE, and Word2Vec, BERT evaluates text in two directions (i.e., left to right and vice-versa). Disaster-based studies incorporating BERT often reported improved classifications; for instance, a series of experiments using seven different catastrophic event datasets based on a multi-modal technique combining BERT and DenseNet yielded promising F-scores ranging from 66% to 88% [22]. The authors of [9] compared BERT-Bi-LSTM with traditional context-free embedding techniques using a Twitter dataset, and found the former exhibited the best results in prediction, with an F-score of 83%. The authors of [9] further extended their analysis to include more embedding techniques, such as GloVe, Skip-Gram, FastText, and other DNN models (RNN, CNN). Results generally indicate that BERT-based modeling yields the best results for disaster-prediction tasks [25]. Other studies implementing BERT include the detection of tweets linked to the Jakarta flood in 2020 [29], COVID-19 crisis communications [24], public datasets, such as crisisLexT26 and crisisNLP [23], etc.

Although BERT often yields good results on NLP tasks, it is, however, resource intensive [30]. Therefore, researchers began to explore simpler versions of BERT (or its variants) such as RoBERTa, TinyBERT, ELECTRA, and AlBERT, among others. However,

a search of the disaster detection studies revealed very few studies that have utilized these variants. For instance, the authors of [14] proposed an ensemble-based strategy by combining RoBERTa, BERTweet, and CT-BERT models to detect COVID-19 related tweets and found their approach to produce the best F-score of 91%, outperforming the traditional machine learning and deep learning algorithms. The authors of [8], on the other hand, proposed a sentiment-aware contextual model named SentiBERT-BiLSTM-CNN for tweet-based disaster detection, with SentiBERT specifically used to extract sentimental contextual embeddings from a given tweet. The authors found their proposed model to outperform the rest of the models, with an F-score of 92.7%.

Table 1 provides the review of deep learning studies discussed in this section. In summary, the review shows that the majority of the studies on Twitter disaster detection were based on the traditional context-free embedding techniques, whereas those exploring the more robust transformer-based techniques were scant. Further, it can be observed that BERT remains to be largely explored in this regard, despite the promising results and performance of its variants [8,14]. The review, therefore, provides an insight into the gaps of AI-based Twitter disaster detection and helps to guide this study in exploring the transformers through deep learning models.

Table 1. Summary of deep learning studies in detecting tweets related to disasters.

References	Disaster	Algorithm	Word Embedding	F-Score (%)
[19]	Hurricane	CNN	Word2Vec	80
[20]	Earthquake	CNN	Bag-of-words, GloVe	96
[21]	Hurricane, wildfire, earthquake, flood	LSTM, CNN	GloVe	93
[31]	26 disasters	LSTM, bi-directional GRU	WordNet	79–82
[2]	Hurricane, flood, wildfire	CNN	GloVe, FastText	71
[27]	Earthquake	MLP	Word2vec	85
[24]	Earthquake, wildfire, flood	CNN	BERT, DenseNet	66–88
[8]	General (i.e., flood, fire, earthquake)	Bi-LSTM–CNN	SentiBERT	92.7
[23]	26 disasters	LSTM, CNN	BERT	71.86
[9]	General (i.e., flood, fire, earthquake)	Bi-LSTM	BERT	83.16
[14]	COVID-19 related disaster	-	RoBERTa, BERTweet, CT-BERT	91

CNN: Convolutional Neural Network; LSTM: Long Short-Term Memory; MLP: Multilayer Perceptron.

3. Materials and Method

Figure 1 depicts a general overview of the pipeline for disaster detection with several consecutive modules.

The pre-processed tweets act as input to the transformers, where contextual word embedding takes place. The output of the transformers are then fitted to a deep NN algorithm to form ensemble models, specifically NN, CNN, LSTM, and Bi-LSTM. The ensemble combination of the transformer and NN models then makes the final prediction, that is, whether a tweet is disaster or non-disaster related. These modules are elaborated in the subsequent sections.

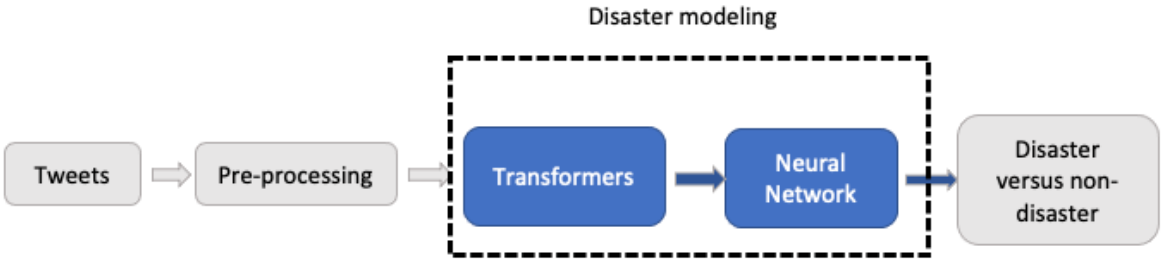

Figure 1. Disaster detection pipeline.

3.1. Twitter Dataset

A Twitter dataset available on Kaggle (https://www.kaggle.com/c/nlp-getting-started/data?select=train.csv, accessed on 23 December 2021) containing 7613 tweets regarding disasters was used in this study. The metadata included ID, keyword (i.e., unique words from the tweet), location (i.e., origin of tweet), and the actual text. The dataset also contained human labels identifying if the tweet pertains to a disaster or otherwise (i.e., binary labels). Specifically, the tweets were classified as 1 (i.e., disaster) or 0 (i.e., not a disaster), with examples of disasters communicated in the dataset including floods, storms, earthquakes, and fires. Table 2 provides some examples for both the disaster and non-disaster tweets. The dataset has been used by the authors of [8,9,25], as stated in Section 2.

Table 2. Sample tweets from the Kaggle disaster dataset (1—Disaster; 0—Non-disaster).

Original Tweets	Label
Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all	1
#Flood in Bago Myanmar #We arrived Bago	1
Forest fire near La Ronge Sask. Canada	1
I love fruits	0
My car is so fast	0
Summer is lovely	0

A simple exploratory data analysis revealed the dataset to be balanced, that is, 42.9% (n = 3271) and 57.1% (n = 4342) for disaster and non-disaster, respectively. The average length of tweets was 12.5 words, with most of the disaster and non-disaster tweets ranging between 10 and 20 words. Analysis also shows that the disaster tweets are relatively longer than the non-disaster tweets [8]. Figure 2 shows an overview of the top words associated with the two labels. It can be observed that most of the disaster-related words, such as fire, storm, death, and flood, were found in the disaster tweets, while the other tweets contain more commonly used words, such as going, love, new, etc. Some disaster-related words, such as fire, burning, and blown, were found in both the labels, albeit with different frequencies. This clearly indicates the possibility of the words having different contextual meanings, hence, the importance of understanding them through the use of contextual word embedding techniques.

3.2. Data Pre-Processing

The next stage involved pre-processing the tweets in order to reduce the "noise" in the social media data. This included the removal of hashtags, punctuation marks, special characters, and stop words, among others. Further, all upper-cases were converted into lower-cases, similar to the methods used in [8,9,25]. The pre-processed tweets served as an input to the data modeling stage (see Figure 1), specifically, for the contextual word embedding (i.e., transformers).

Figure 2. Overview of the top words for disaster and non-disaster tweets.

3.3. Contextual Word Embedding (Transformers)

Six transformer-based contextual word embedding techniques were examined in the current study, including BERT (both the base/small and large variants), ELECTRA, Bert Expert, Talking-Heads Attention, and TN-Bert. The BERT is the original transformer, and is a base model (i.e., other variants were extended/modified from this) (see Figure 3). There are two main variants in BERT, BERT-base/small and BERT-large, consisting of 12 and 24 transformer blocks, respectively. BERT is a multi-head attention-based (i.e., each head performs separate computations that are aggregated at the end) language model that employs the transformer encoders and decoders to learn the contextual relationships between words [28]. The encoder reads the text input, while the decoder produces a prediction. In BERT, the bidirectional transformer NN acts as the encoder, converting each tokenized word into its numerical vector in order to translate words that are semantically related to embedding that is numerically close [8]. It specifically uses the Wordpiece embedding input for tokens, along with positional (i.e., the position of each token in a given sequence) and segment (i.e., when sentence pairs are used as input) embedding for each token. The final embedding is usually the sum of all the embedding (i.e., token, positional, and segment). BERT uses the masked language modeling (MLM) approach, in which tokens are randomly replaced with [MASK], and a model is trained to reconstruct the tokens that have been replaced. The embedded numerical vectors are then fed into the Softmax, which makes a final prediction. In this study, however, the Softmax layer is replaced with a deep NN model that makes the final prediction, in line with previous studies [9,22]. Although BERT has been shown to produce good results on NLP tasks, it is however, impractical for use on resource-limited devices, as it is computationally expensive [26,30]. This resulted in the emergence of BERT variants, such as TN-Bert, which is a compressed version of the original BERT architecture, using tensor networks. Previous experiments have shown the variant to be 37% smaller and 22% faster than BERT-base [32].

On the other hand, ELECTRA is identical to BERT, except there has an additional linear layer between the embedding layer and encoder. Unlike other models that are based on MLM pre-training, ELECTRA is a discriminator that replaces random tokens with fake tokens, akin to the technique adopted in the generative adversarial network (GAN), in which a generator is optimized to train the discriminator [30]. This approach is considered less costly and more efficient. In fact, ELECTRA is often touted to be one of the best variants, performing with a fraction of the computing power of BERT [30].

Other lesser-known variants include Talking Head, or Talking-Heads Attention, which is a new variation on the multi-head attention used in BERT, using linear projections across the attention-heads, before and after the Softmax operation. This variant has been shown to have better performance in MLM tasks, as well as question/answer tasks, despite having a small number of additional parameters and computation ability [33]. On the other hand, the BertExpert was developed using a fine-tuned collection of BERTs that are trained on eight different datasets, comprising six Wikipedia and BooksCorpus datasets and two

PubMed datasets, to improve its performance in the NLP [28]. The pre-trained word embedding produced by the transformers is then used as input to the NN pipeline for disaster detection.

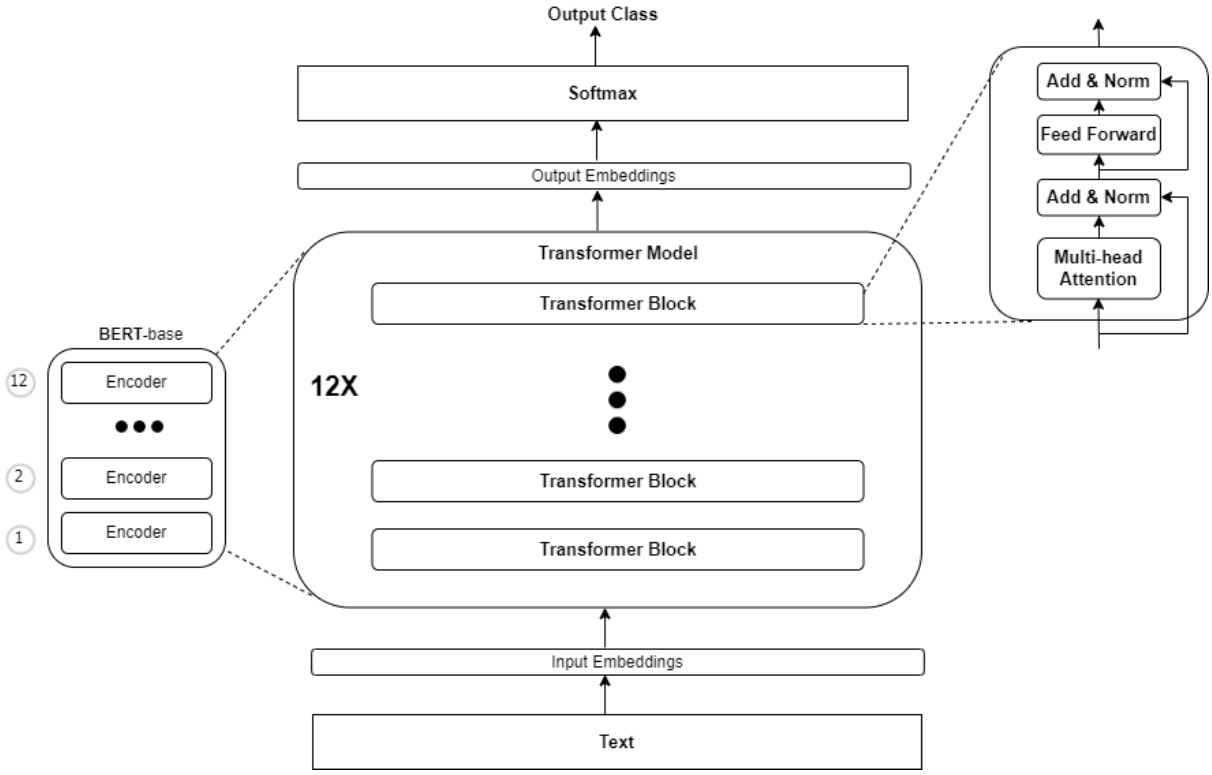

Figure 3. BERT-base architecture.

It is worth noting that since the variants were based on the BERT-base model, the word embedding technique does not differ from the base model; instead, the variants mainly differ in terms of simplicity (e.g., reduced number of layers). For example, the only difference regarding ELECTRA is the separation of the embedding size and the hidden size, compared to BERT [30].

3.4. Disaster Modeling

As the main focus of this paper is on the contextual word embedding techniques, this section presents only a brief overview of the DNN algorithms used. NN refers to a network in which all neurons in a layer are fully connected by weighted links to other neurons in the next layers. Inspired by the biological nervous system, NN generally has three layers, namely, input (i.e., receives and presents input pattern to the network), hidden (optional) (i.e., transforms input inside the network), and output (i.e., returns value corresponding to the prediction of the response variable). Activation functions define how the weighted sum of the input is transformed into an output, with popular functions, including Rectified Linear Activation (ReLU), Sigmoid, and Tanh, for the hidden layer, and Sigmoid and Softmax for the output layer. In this paper, the output of the transformer (e.g., BERT) encoders are fitted to NN with the following sequence of layers: a layer of 32 neurons, a dropout layer, and an output layer, with Sigmoid as the activation function [34–36]. It is worth noting that the NN model was used as a baseline for comparison with the DNN models.

CNN consists of convolution layers, a pooling layer, and fully connected output layers. The convolution layers apply filters with a specific kernel size to learn features from a given dataset, whereas the pooling layer serves as an intermediate layer to reduce the dimensions of the convolution layers output. The output layer contains activation functions to predict

the final class of the input dataset [37]. We used a convolution layer with 256 filters and a window size of 3, 4, and 5-word vectors, with a kernel regularizer that applies an L1 regularization penalty with a value of 0.01, along with ReLU as the activation function, a pooling layer with max pooling (i.e., pool size = 4), and an output layer using Sigmoid.

On the other hand, LSTM can be viewed as an improvised version of RNN, consisting of a set of recurrently connected blocks (i.e., memory blocks) with three gates, that is, an input gate, a forget gate, and an output gate [38]. It is capable of learning order-dependence in sequence-prediction problems. We used an LSTM layer with 256 units and ReLU as the activation function. Sigmoid, on the other hand was used as the activation function for the output layer. The Bi-LSTM is similar to the LSTM; however, it processes input data in forward and backward directions [38]. The Bi-LSTM layer consists of 128 neurons, whereas the dense layer consists of 64 neurons. Tanh and ReLU were used as the activation functions for these two layers, respectively. Sigmoid was used for the output layer. All the NN models were executed using Adam as the optimizer, with a learning rate of 0.00003, and binary cross entropy as the loss function. Sigmoid was selected as the activation function for all the models, considering that the final prediction is based on binary labels. Table 3 shows the parameter setups used for all the models.

Table 3. Parameter setups for DNN models.

Models	Setup	Parameters
NN	Layers	3 (Neurons = 32)
	Dropout rate	0.1 *
	Activation Function	ReLU (Hidden) & Sigmoid (Output)
CNN	Layers	3 (Filters = 256; Kernel: 3–5)
	Dropout rate	0.3
	Activation Function	ReLU (Hidden) & Sigmoid (Output)
LSTM	Layers	3 (Neuron = 256)
	Dropout rate	0.3
	Activation Function	Tanh (Hidden) & Sigmoid (Output)
Bi-LSTM	Layers	LSTM(units = 128, activation = "tanh") Dense (units = 64, activation = "ReLU") Dense(units = 1, activation = "sigmoid")
	Dropout rate	0.3
	Activation Function	Tanh (LSTM), ReLU & Sigmoid

* Note: All the dropout rates were determined using the grid search approach, hence, the dissimilarity between NN and the rest of the models.

3.5. Evaluation and Experiments

Considering that Twitter disaster detection is a classification problem (i.e., disaster versus non-disaster), standard classification metrics were used to assess the effectiveness of the proposed models' performance [8,9,25]. These include precision, recall, F-score, accuracy and area under curve (AUC). As per the tweet labels that are binary in nature, a disaster (i.e., 1) is deemed as a positive class, while a non-disaster is a negative class. Therefore, true positive (TP) refers to the actual disaster tweets predicted as disasters, whereas false positive (FP) refers to tweets that are actually false, but predicted as true. A similar implication applies to true negative (TN) and false negative (FN). Using these notations, accuracy refers to the number of correctly predicted tweets among all of the tweets, and can be denoted using Equation (1) below:

$$Accuracy\ (Acc) = \frac{TP + TN}{TP + FN + TN + FP} \quad (1)$$

where TP—true positive; TN—true negative; FN—false negative; TN—true negative.

Precision reflects the proportion of TP over the total sample, whereas recall is the number of positive classes missed (i.e., proportion of the correctly identified TP over the sample predicted as positive by the model) [8,9]. Both these metrics can be determined using Equations (2) and (3) below.

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{2}$$

$$Recall\ (R) = \frac{TP}{TP + TN} \tag{3}$$

F-score refers to the harmonic mean of recall and precision, and determined using Equation (4) below.

$$F - score = \frac{2 \cdot Precision \cdot Recall}{Precision \cdot Recall} \tag{4}$$

Finally, AUC provides an indication as to how well a model is capable of distinguishing between classes, with higher scores meaning the model works better at predicting positive classes as 1 (disaster) and negative classes as 0 (non-disaster). It can be determined using Equation (5), as given below:

$$AUC = \frac{S_p - \frac{n_p(n_n+1)}{2}}{n_p n_n} \tag{5}$$

where S_p indicates the sum of all positive samples, n_p indicates the number of positive examples, and n_n indicates the number of negative samples.

All five metrics return a score between 0 and 1, with a higher score indicating a better detection/classification performance.

The training and testing were accomplished using an 80–20 split. Table 4 below depicts the descriptive statistics for the split data used for training (i.e., 80% = 6090). The average length of tweets was 14.9, while 7273 words have frequency >1. Figure 4 shows the word length distribution for disaster and non-disaster tweets. It can be observed that the disaster tweets are generally longer than non-disaster tweets, although most of them are between a word length of 10 to 20.

Table 4. Descriptive statistics for 80% training data.

Characteristics	*n*
Total training data	6090
Total positive data (or disaster tweets)	2617
Total unique words	27,083
Total unique words with frequency >1	7253
Avg. length of tweets	14.9
Median length of tweets	15.0
Maximum length of tweets	31
Minimum length of tweets	1

All the experiments and modeling were accomplished using Python 3.7.12 (with sklearn library) and TensorFlow 2.7.0, with a GPU NVIDIA Tesla P100.

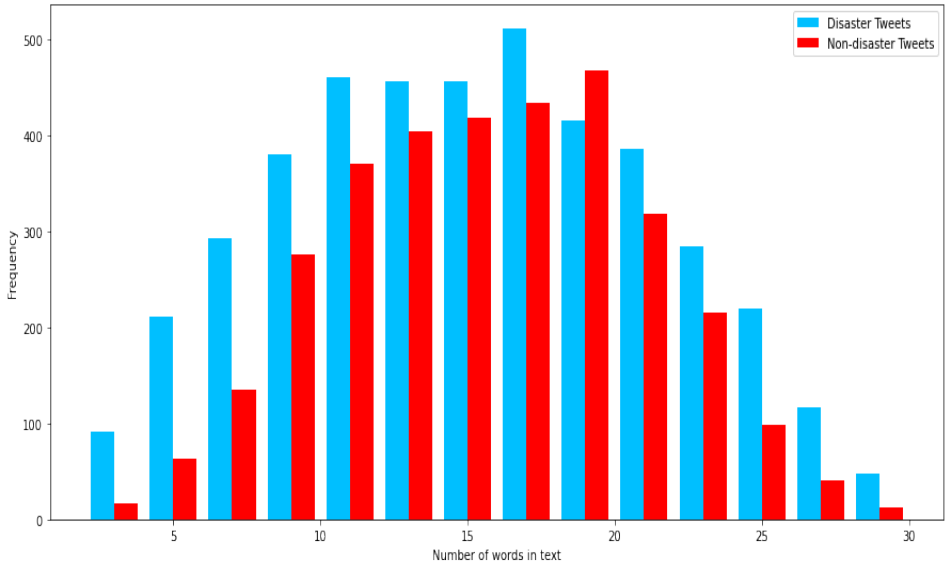

Figure 4. Word length distribution for disaster and non-disaster tweets (training data).

4. Results and Discussion

Table 5 depicts the results of the experiments for each of the ensemble models.

Table 5. Ensemble model performance results.

Model	Transformers	Accuracy	Precision	Recall	F-Score	AUC
NN	$BERT_{Large}$	0.82	0.83	0.73	0.78	0.86
	$BERT_{Small}$	0.82	0.81	0.76	0.78	0.88
	ELECTRA	0.83	0.80	0.79	0.79	0.89
	TN-BERT	0.82	0.87	0.69	0.77	0.88
	BERT Expert	0.83	0.81	0.77	0.79	0.89
	Talking Head	0.83	0.86	0.71	0.78	0.88
LSTM	$BERT_{Large}$	0.82	0.88	0.67	0.76	0.88
	$BERT_{Small}$	0.81	0.87	0.67	0.76	0.88
	ELECTRA	0.83	0.85	0.74	0.79	0.89
	TN-BERT	0.82	0.88	0.67	0.76	0.87
	BERT Expert	0.81	0.76	0.81	0.79	0.89
	Talking Head	0.82	0.84	0.72	0.77	0.88
CNN	$BERT_{Large}$	0.83	0.89	0.68	0.77	0.88
	$BERT_{Small}$	0.79	0.72	0.83	0.77	0.88
	ELECTRA	0.82	0.78	0.82	0.80	0.89
	TN-BERT	0.82	0.92	0.64	0.75	0.87
	BERT Expert	0.84	0.91	0.69	0.78	0.89
	Talking Head	0.83	0.87	0.72	0.79	0.88

Table 5. *Cont.*

Model	Transformers	Accuracy	Precision	Recall	F-Score	AUC
BiLSTM	$BERT_{Large}$	0.83	0.87	0.70	0.78	0.88
	$BERT_{Small}$	0.80	0.79	0.74	0.76	0.88
	ELECTRA	0.82	0.79	0.81	0.80	0.90
	TN-BERT	0.83	0.84	0.74	0.79	0.88
	BERT Expert	0.81	0.75	0.83	0.79	0.89
	Talking Head	0.84	0.87	0.74	0.80	0.89

The results generally indicate that all the models consistently perform well, with F-scores ranging from 76% to 80%, and accuracy scores from 79% to 83%. A similar observation was noted for AUC, which ranged between 86% and 90%. This supports previous findings showing that transformer-based contextual word embedding techniques improve disaster detection on social media [8,9,14,23–25]. This is probably due to the nature of the transformers, that is, the techniques take context of words into consideration, hence their ability to interpret ambiguous words in a sentence supersede the context-free embedding techniques, such as Word2Vec and GloVe [2,19,21].

Although the performance scores are only marginally different, an overall comparison between the ensembles revealed Bi-LSTM to yield the best performance across all five metrics, with scores ranging from 70% to 90%. Bi-LSTM is deemed to be an improvement over the previous DNN models, including LSTM (which is an improvement of RNN). Although this does not necessarily result in better detection performance, the bidirectional text processing employed by Bi-LSTM may also have contributed to its good performance, akin to BERT and its variants. A similar pattern was reflected by the authors of [9], who worked on the same dataset using BERT-small, for which the authors reported a slightly better performance with an F-score of 83% and accuracy of 85.6%.

Interestingly, the simpler and less complicated BERT variants are observed to yield comparable, and even superior results, to the original BERT (i.e., $BERT_{Large}$ and $BERT_{Small}$). The top two variants that yielded consistently good performance across all the metrics and NN models include ELECTRA and Talking Head, with the best performance noted in combination with Bi-LSTM ($F\text{-score}_{ELECTRA}$ = 80%; $Accuracy_{ELECTRA}$ = 82%; $F\text{-score}_{TalkingHead}$ = 80%; $Accuracy_{TalkingHead}$ = 84%). Table 6 provides a few sample tweets and the classification results for both of these models. To the best of our knowledge, these variants have yet to be explored by other researchers, including those focusing on disaster detection and management. Nevertheless, our findings, support the results reported by the respective developers, in which the variants were notably found to outperform most of the transformers, including the original BERT [29,32]. This finding is deemed novel, and promises to offer a feasible and cost-effective solution in detecting disasters via social media, without requiring intensive and expensive resources.

Table 6. Sample detection results for Bi-LSTM.

Sample Tweets	Prediction		True Label
	Talking Head	ELECTRA	
The summer program I worked for went the city pool we had to evacuate because one of my kids left a surprise. @jimmyfallon #WorstSummerJob	1	0	0
You are the avalanche. One world away. My make believing. While I'm wide awake.	0	0	0
Dorman 917-033 Ignition Knock (Detonation) Sensor Connector http://t.co/WxCes39ZTe http://t.co/PyGKSSSCFR	0	0	1

Table 6. *Cont.*

Sample Tweets	Prediction		True Label
	Talking Head	ELECTRA	
Christian Attacked by Muslims at the Temple Mount after Waving Israeli Flag via Pamela Geller- ... http://t.co/wGWiQmICL1	1	1	1
70 Years After Atomic Bombs Japan Still Struggles With War Past: The anniversary of the devastation wrought b ... http://t.co/vFCtrzaOk2	1	1	1
You are equally as scared cause this somehow started to heal you fill your wounds that you once thought were permanent.	0	0	0
@abcnews UK scandal of 2009 caused major upheaval to Parliamentary expenses with subsequent sackings and prison. What are we waiting for?	0	0	0
Expect gusty winds heavy downpours and lightning moving northeast toward VA now. http://t.co/jyxafD4knK	1	1	1
August 5: Your daily horoscope: A relationship upheaval over the next few months may be disruptive but in the ... http://t.co/gk4uNPZNhN	0	0	0
@BattleRoyaleMod when they die they just get teleported into somewhere middle of ocean and stays trapped in there unless they decides 2/6	0	0	0

5. Conclusions, Limitation and Future Direction

This study explored the use of transformer-based contextual word embedding with deep NN models to detect disaster-related communication on Twitter. The popular BERT model, along with its lesser-known variants, were explored by combining them with CNN, NN, LSTM, and Bi-LSTM. Experimental results show all the ensemble models to yield consistently good results, with F-scores ranging from 76% to 80%. Although only marginally different, ELECTRA and Talking Head variants produced the best results when combined with Bi-LSTM. Our results added value to the existing literature, as we showed that disaster detection is effective and efficient with the use of simpler and less complicated variants.

This study only used a single Twitter dataset (n = 7613). Additional experiments and analyses involving a larger dataset, and comparisons with similar datasets, including those from other social media platforms, such as Facebook, would be beneficial. This will help to establish the performance of the variants and NN models in detecting disaster through textual communications. Further, the study is also limited to communications in English. Social media platform users are diversified, with communication taking place in various languages, including Chinese, Hindi, and French, among others. Transformers, such as BERT, provide support in processing multi-language texts; hence, future studies should explore the performance of these techniques by using datasets that are not limited to English. Finally, the study is also limited by using a single modality (i.e., text). Communications on social media platforms often include images and videos as well; therefore, future studies can explore detecting disaster-related communication using a multi-modal input.

Author Contributions: Conceptualization, V.B.; Formal analysis, C.L.L. and R.L.; Funding acquisition, J.P.; Methodology, Z.S., L.L.T. and Y.F.; Project administration, V.B.; Draft revision: V.B., C.L.L. and Z.S.; Writing—original draft, V.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are available on Kaggle (https://www.kaggle.com/c/nlp-getting-started/data?select=train.csv, accessed on 23 December 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. DiCarlo, M.F.; Berglund, E.Z. Connected communities improve hazard response: An agent-based model of social media behaviors during hurricanes. *Sustain. Cities Soc.* **2021**, *69*, 102836. [CrossRef]
2. Roy, P.K.; Kumar, A.; Singh, J.P.; Dwivedi, Y.K.; Rana, N.P.; Raman, R. Disaster related social media content processing for sustainable cities. *Sustain. Cities Soc.* **2021**, *75*, 103363. [CrossRef]
3. Rhodan, M. Please Send Help: Hurricane Harvey Victims Turn to Twitter and Facebook. 2017. Available online: http://time.com/4921961/hurricane-harvey-twitter-facebook-social-media/ (accessed on 13 February 2022).
4. Son, J.; Lee, H.K.; Jin, S.; Lee, J. Content features of tweets for effective communication during disasters: A media synchronicity theory perspective. *Int. J. Inf. Manag.* **2019**, *45*, 56–68. [CrossRef]
5. Zhai, W.; Peng, Z.R.; Yuan, F. Examine the effects of neighborhood equity on disaster situational awareness: Harness machine learning and geotagged Twitter data. *Int. J. Disaster Risk Reduct.* **2021**, *48*, 101611. [CrossRef]
6. Karimiziarani, M.; Jafarzadegan, K.; Abbaszadeh, P.; Shao, W.; Moradkhani, H. Hazard risk awareness and disaster management: Extracting the information content of twitter data. *Sustain. Cities Soc.* **2022**, *77*, 103577. [CrossRef]
7. Robertson, B.W.; Johnson, M.; Murthy, D.; Smith, W.R.; Stephes, K.K. Using a combination of human insights and 'deep learning' for real-time disaster communication. *Prog. Disaster Sci.* **2019**, *2*, 100030. [CrossRef]
8. Song, G.; Huang, D.A. Sentiment-Aware Contextual Model for Real-Time Disaster Prediction Using Twitter Data. *Future Internet* **2021**, *13*, 163. [CrossRef]
9. Chanda, A.K. Efficacy of BERT embeddings on predicting disaster from Twitter data. *arXiv* **2021**, arXiv:2108.10698.
10. Chen, Z.; Lim, S. Social media data-based typhoon disaster assessment. *Int. J. Disaster Risk Reduct.* **2021**, *64*, 102482. [CrossRef]
11. Resch, B.; Usländer, F.; Havas, C. Combining machine-learning topic models and spatiotemporal analysis of social media data for disaster footprint and damage assessment. *Cartogr. Geogr. Inf. Sci.* **2018**, *45*, 362–376. [CrossRef]
12. Ragini, J.R.; Anand, P.R.; Bhaskar, V. Big data analytics for disaster response and recovery through sentiment analysis. *Int. J. Inf. Manag.* **2018**, *42*, 13–24. [CrossRef]
13. Neppalli, V.K.; Caragea, C.; Squicciarini, A.; Tapia, A.; Stehle, S. Sentiment analysis during hurricane sandy in emergency response. *Int. J. Disaster Risk Reduct.* **2017**, *21*, 213–222. [CrossRef]
14. Malla, S.; Alphonse, P. COVID-19 outbreak: An ensemble pre-trained deep learning model for detecting informative tweets. *Appl. Soft Comput.* **2021**, *107*, 107495. [CrossRef]
15. Nazer, T.H.; Morstatter, F.; Dani, H.; Liu, H. Finding requests in social media for disaster relief. In Proceedings of the 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), San Francisco, CA, USA, 18–21 August 2016; pp. 1410–1413.
16. Alam, F.; Ofli, F.; Imran, M. Descriptive and visual summaries of disaster events using artificial intelligence techniques: Case studies of Hurricanes Harvey, Irma, and Maria. *Behav. Inf. Technol.* **2019**, *39*, 288–318. [CrossRef]
17. Basu, M.; Shandilya, A.; Khosla, P.; Ghosh, K.; Ghosh, S. Extracting resource needs and availabilities from microblogs for aiding post-disaster relief operations. *IEEE Trans. Comput. Soc. Syst.* **2019**, *6*, 604–618. [CrossRef]
18. Mohanty, S.D.; Biggers, B.; Ahmed, S.S.; Pourebrahim, N.; Goldstein, E.B.; Bunch, R.; Chi, G.; Sadri, F.; McCoy, T.; Cosby, A. A multi-modal approach towards mining social media data during natural disasters—A case study of Hurricane Irma. *Int. J. Disaster Risk Reduct.* **2021**, *54*, 102032. [CrossRef]
19. Yu, M.; Huang, Q.; Qin, H.; Scheele, C.; Yang, C. Deep learning for real-time social media text classification for situation awareness–using hurricanes sandy, harvey, and irma as case studies. *Int. J. Digit. Earth* **2019**, *12*, 1230–1247. [CrossRef]
20. Kumar, A.; Singh, J.P. Location reference identification from tweets during emergencies: A deep learning approach. *Int. J. Disaster Risk Reduct.* **2019**, *33*, 365–375. [CrossRef]
21. Kumar, A.; Singh, J.P.; Dwivedi, Y.K.; Rana, N.P. A deep multi-modal neural network for informative twitter content classification during emergencies. *Ann. Oper. Res.* **2020**, 1–32. [CrossRef]
22. Madichetty, S.; Muthukumarasamy, S.; Jayadev, P. Multi-modal classification of twitter data during disasters for humanitarian response. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 10223–10237. [CrossRef]
23. Naaz, S.; Ul-Abedin, Z.; Rizvi, D.R. Sequence Classification of Tweets with Transfer Learning via BERT in the Field of Disaster Management. *EAI Endorsed Trans. Scalable Inf. Syst.* **2021**, *8*, e8. [CrossRef]
24. Wang, Z.; Zhu, T.; Mai, S. Disaster Detector on Twitter Using Bidirectional Encoder Representation from Transformers with Keyword Position Information. In Proceedings of the 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology, Weihai, China, 14–16 October 2020; pp. 474–477.
25. Deb, S.; Chanda, A.K. Comparative analysis of contextual and context-free embeddings in disaster prediction from Twitter data. *Mach. Learn. Appl.* **2022**, *7*, 100253. [CrossRef]
26. Qui, X.; Sun, T.; Xu, Y.; Shao, Y.; Huang, X. Pre-trained Models for Natural Language Processing: A Survey. *arXiv* **2021**, arXiv:2003.08271.
27. Behl, S.; Rao, A.; Aggarwal, S.; Chadha, S.; Pannu, H.S. Twitter for disaster relief through sentiment analysis for COVID-19 and natural hazard crises. *Int. J. Disaster Risk Reduct.* **2021**, *55*, 102101. [CrossRef]

28. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. *BERT: Pre-Training of Deep Bidirectional Transformers for BERT: Pre-training of Deep Bidirectional Transformers for*; NAACL-HLT 2019; Association for Computational Linguistics: Minneapolis, MI, USA, 2019; pp. 4171–4186.
29. Maharani, W. Sentiment Analysis during Jakarta Flood for Emergency Responses and Situational Awareness in Disaster Management using BERT. In Proceedings of the 2020 8th International Conference on Information and Communication Technology (ICoICT), Yogyakarta, Indonesia, 24–26 June 2020; pp. 1–5.
30. Clark, K.; Luong, M.T.; Le, Q.V.; Manning, C.D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *arXiv* **2020**, arXiv:2003.10555.
31. Bhuvaneswari, A.; Thomas, J.T.J.; Kesavan, P. Embedded Bi-directional GRU and LSTM Learning Models to Predict Disasters on Twitter Data. *Procedia Comput. Sci.* **2019**, *165*, 511–516. [CrossRef]
32. Abadi, M.; Ashish, A.; Barham, P.; Eugene, B.; Chen, Z.; Davis, A.; Dean, J. TensorFlow: TN_BERT. 2021. Available online: https://tfhub.dev/google/tn_bert/1 (accessed on 28 December 2021).
33. Shazeer, N.; Lan, Z.Z.; Cheng, Y.; Ding, N.; Hou, L. Talking Heads Attention. *arXiv* **2020**, arXiv:2003.02436v1.
34. Zhao, Y.; Ren, S.; Kurthscde, J. Synchronization of coupled memristive competitive BAM neural networks with different time scales. *Neurocomputing* **2021**, *427*, 110–117. [CrossRef]
35. Alqatawna, A.; Álvarez, A.M.R.; García-Moreno, S.S.C. Comparison of Multivariate Regression Models and Artificial Neural Networks for Prediction Highway Traffic Accidents in Spain: A Case Study. *Transp. Res. Procedia* **2021**, *58*, 277–284. [CrossRef]
36. Bre, F.; Gimenez, J.; Fachinotti, V. Prediction of wind pressure coefficients on building surfaces using Artificial Neural Networks. *Energy Build.* **2018**, *158*, 1429–1441. [CrossRef]
37. Zhang, L.; Wang, S.; Liu, B. Deep learning for sentiment analysis: A survey. *WIREs Data Min. Knowl. Discov.* **2018**, *8*, e1253. [CrossRef]
38. Jang, B.; Kim, M.; Harerimana, G.; Kang, S.-U.; Kim, J. Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism. *Appl. Sci.* **2020**, *10*, 5841. [CrossRef]