Current Applied Science and Technology e-ISSN: 2586-9396

Curr. Appl. Sci. Technol. 2025, Vol. 25 (No. 1), e0259284

Research article

ABiLSTM with BERT Embedding for Classification of Imbalanced COVID-19 Rumors

Rakesh Dutta[1] and Mukta Majumder[2]*

[1]*Department of Computer Science and Application, Hijli College, Kharagpur, India*
[2]*Department of Computer Science and Technology, University of North Bengal, Siliguri, India*

Received: 20 June 2023, Revised: 25 March 2024, Accepted: 27 June 2024, Published: 17 October 2024

Abstract

The coronavirus emerged at the end of 2019 and has caused thousands of casualties all over the world. The pandemic has also been accompanied by loss of employment and economic down fall. Naturally, the pandemic and lack of knowledge of coronavirus has created public anxiety and panic. Nowadays, social medias like Twitter and Facebook and online news forum reach most people and have become popular channels of communication and information sharing. Unfortunately, these have become easy targets for rumors and fake news. The rapid flow of rumors and misleading information on the coronavirus over these online platforms has promoted public anxiety and fear. Consequently, the detection of rumors has become obligatory for economy and public safety. In this context, the present research focused on detecting and classifying rumors so that precautionary measures can be incorporated. Attention-based BiLSTM with BERT for rumor classification on the COVID-19 rumor dataset was proposed. The suggested classification model achieved an accuracy of 80.71% and a micro-F1 score of 90.85. Furthermore, the experimental outcomes affirm the superior efficacy of our proposed technique over existing methods.

Keywords: BERT; BiLSTM; attention mechanism; word embedding; rumor classification; COVID-19

1. Introduction

Apart from health crises, the COVID-19 pandemic has caused serious economic, employment, educational, and socio-psychological effects on a global scale. During the lockdown in the viral outbreak, public were forced to remain within the walls of their homes. A number of studies have shown rapid growth of addiction of common public towards smart phones, social medias and online blogs/forums during the pandemic (Nguyen et al., 2017; Bratu, 2020; Hui et al., 2020). They use these online platforms to share information and find support or consolation. Unfortunately, such information is not generally reliable; it often incorporates rumor, misinformation, and false news. This sort of information is flooding the

*Corresponding author: E-mail: mukta_jgec_it_4@yahoo.co.in
https://doi.org/10.55003/cast.2024.259284

internet. The information has been shown to encourage individuals to cast their opinions (Mihaylov et al., 2015a, b; Mihaylov & Nakov, 2016), deceive people through frauds money transactions via link clicks (Bourgonje et al., 2017), and influencing major events (Vosoughi et al., 2018). The rumors are deliberately produced by humans and disseminated through these online channels. People's daily lives have been severely damaged since the COVID-19 outbreak and now another type of virus 'rumor' has emerged. Throughout the pandemic, generation and dissemination of rumors have occurred rapidly and have affected our socio-psychological and economic conditions (DiFonzo & Bordia, 2007). These rumors have hampered pandemic prevention and control, social governance and so on. Some rumors have seriously misled individuals, resulting in risk to life, health crises, and social instability (Hui et al., 2020). Rumors are frequently disseminated via social media, news blogs (Sahni & Sharma, 2020). Due to public interest, every social networking newscast must adhere to a criterion of verifiability. Although electronic media have made significant strides, there are still many challenges to be faced in the control of the propagation of confusing and false news and rumors. The rapid propagation of rumors has often been seen in the devastating worldwide outbreak of the COVID-19 pandemic (Bratu, 2020). The majority of people claim that they have been exposed with misleading rumors on social media frequently (Bratu, 2020). People are being victimized and cheated under the influence of topic-related rumors, misleading information, and false news. These types of deception have often led individuals to make rash decisions which have caused them to become panic-stricken, aggressive, and racist. As rumors are spread out in disguised to confuse viewers and listeners; making them challenging to detect and classify. System-defined rumor detection has become a significant area of research in light of increasing technical breakthroughs. It is essential to choose a suitable strategy for detecting COVID-19 rumors successfully. Such attempts require correct categorization, association, and correlation approaches for distinguishing social deception criteria, such as right information or misinformation.

In response to this problem, this study was attempted to classify rumors related to COVID-19 from popular news providers such as BBC News, CNN, Reuters, ABC News and Twitter. An attention based BiLSTM with BERT model (BERT+ABiLSTM) for the task was designed. The model capable of obtaining richer semantic information by incorporating BERT hidden unit was proposed.

2. Materials and Methods

2.1 Materials

Rumor detection is an innovative research area based on natural language processing and is similar to fake news recognition. Although rumors are distinct from fake news, and often consist in false information that is passed on by someone who is unaware of the truth or otherwise of the information. The information may be true in part or false whereas fake news is a total fabrication that is made to look like real news. In this section, existing related works were reviewed.

2.1.1 Rumors and problem

People and journalists are very worried about rumors and fake news. These can cause economic loss, and reduction in public trust on administration, marketing, and broadcasting media. False news deceives people (Long et al., 2017). In August 2015, rumors about the kidnapping of children by drug gangs spread throughout Veracruz via Twitter and Facebook (Ma et al., 2016). These caused 26 car accidents when people rushed to get their kids. Thus, it is very important to be able to predict the truth of social media information automatically (Ma et al., 2016). Detecting fake news and rumors is an intriguing challenge; nevertheless, it can be extremely difficult even for a well-informed person to precisely differentiate between real news, rumors and fake news. Throughout the research, it was noticed that rumor and real news were often combined rather intricately and thus difficult to distinguish (Pham, 2018). Many researchers used a range of machine learning as well as deep learning techniques to lessen their negative impact on society. Initially, we examined research based on machine learning, and the use of deep learning tools was then studied.

2.1.2 Machine learning based studies

Zubiaga et al. (2017) applied Conditional Random Field for rumor detection and classification of the PHEME social media dataset. They compared it to Naïve Bayes (NB), Support Vector Machine (SVM), and Random Forests (RF) classifiers and achieved an f-measure 60.7. Vijeev et al. (2018) applied machine learning techniques to detect rumors based on user and content features. They used the Chi-square test to pick the best features (such as, User_friends_count, User_followers_count, Age_of_tweet, and Retweet_count, etc.) and applied them to the SVM, NB, and RF classifiers to classify the PHEME dataset. They achieved 78% accuracy after combining the features. Zhao et al. (2015) proposed an enquiry phrase-based method to detect rumors in social media by applying Enquiring Minds (a technique involves searching for specific enquiry phrases) and 52% precision was achieved. The main disadvantages of their system were that it required a lot of manual annotation and it was time consuming. Qazvinian et al. (2011) addressed the challenge of rumor recognition in microblogs through the Bayes classifier by analyzing the efficiency of 3 classes of features: network-based, microblog-specific and content-based memes, to accurately identify rumors. Tripathy et al. (2010) applied logistic regression to identify rumors and provided an analysis of ways for controlling rumors in social networks. Yang et al. (2012) used a SVM classifier to look for rumors on Sina Weibo, a popular microblog site in China. This work was based on location and client-based features. Wu et al. (2015) designed a hybrid SVM classifier to receive high-order propagation patterns to detect rumors on Sina Weibo and a classification accuracy of 91.3% was acquired.

2.1.3 Deep learning based techniques

Ma et al. (2016) suggested a model of recurrent neural network (RNN) to learn hidden relationships of the context to identify rumors from the Twitter and Sina Weibo datasets. Their model provided an accuracy of 83.9% on the Twitter dataset and 89.0% on the Weibo dataset. Nguyen et al. (2017) combined convolutional neural network (CNN) and RNN to develop a model for rumor detection which achieved 81.9% accuracy. They scrolled the dataset from online rumor tracking websites for their work (Nguyen et al., 2017). Jin et al. (2017) designed a unique RNN model with an Attention Technique (att-RNN) to combine multimodal characteristics for better rumor classification from Twitter and Sina Weibo social media data by integrating audio and video features. The model delivered an accuracy of 77.8% on the Sina Weibo dataset and accuracy of 62.8% on the Twitter dataset. Guo et al. (2018) developed HSA-BLSTM, an innovative hierarchical neural network for rumor detection on the Twitter and Weibo dataset. Their system obtained accuracies of 84.4% and 94.3% on the Twitter and Weibo dataset, respectively. Alkhodair et al. (2020) used the PHEME dataset to identify rumor from breaking news. For this study, a model consisted of word2vec and LSTM-RNN was built. Sequences of words were transformed into vectors with the help of the word2Vec. Additional vectors were given as an input to the LSTM-RNN model for classifying tweets. The work achieved an accuracy of 79.5% by incorporating content, social, and grammatical features. Asghar et al. (2021) designed a BiLSTM-CNN model for rumor detection. They used the PHEME benchmark dataset and achieved an accuracy of 86.12%. Aker et al. (2019) designed an inner attention LSTM model to check rumors and obtained an f-measure of 0.811 and an accuracy of 81.3%. Guo et al. (2022) proposed a BERT+BiLSTM with Softmax algorithm to analyse and classify user sentiments in message on a “Tree Hole” named “Zou Fan” preceding and following the emergence of COVID-19. The researchers conducted their study to support “Tree Hole” rescue workers as they helped depressed patients, particularly during the COVID-19 outbreak. They found that the number of such messages was positively correlated to emotion in multiple time dimensions. They also found that the bigger the “Tree Hole”, the greater the amount of negative sentiment were there. Yang and Pan (2021) introduced techniques for detecting rumors on social networks related to COVID-19 by leveraging features associated with both the content of the rumor and user responses. This approach was designed to address the swift dissemination and distinctive domain characteristics of COVID-19 rumors on social platforms. They designed a language model using transfer learning and implemented a post-training mechanism to establish CSN-BERT, specifically tailored for COVID-19 user posts on social networks. A comparative analysis was conducted with CR-LSTM-BE, an ensemble deep learning model that integrates user response information into the learning process through LSTM. The experimental findings demonstrated the superior ability of the post-trained CSN-BERT model in extracting content features related to COVID-19 rumors on social networks compared to alternative deep learning models.

2.1.4 BERT language model

The text is first separated into sentence and word-level granularity and then the BERT (Bidirectional Encoder Representation from Transformers) model was employed to generate the text vectors (Li et al., 2019). To appropriately capture the semantic information about context in the classifier, it is important to use the model interface, which represents the embedding of every word in the rumor text. The two-way transformer encoding serves as the primary structural element of the BERT language model. BERT transformer incorporates the self-attention technique and also makes use of the residual technique of the convolutional neural network, resulting in a model with a high training speed and powerful expressiveness ability. Figure 1 depicts the general architecture of BERT model without the RNN loop structure.
In Figure 1, S_n is the encoded expression of each word, V_{e_p} is the architecture of the transformer, and V_n is the representation of target words into vectors after training. The basic idea of the BERT model is to employ the transformer architecture to develop a multi-layer bidirectional encoding network capable of reading the complete text sequence at once and integrating contextual information at each layer. The input of the BERT model is the addition of three embedding methods, Token Embeddings, Segmentation Embeddings, and Position Embeddings, which are used for pre-training and anticipate the subsequent sentence. In text processing, the different semantic meanings of words in the context are based on the position of the word in the context. The transformer specifies that the embedded information of the text is either its relative position or its absolute position, as demonstrated by the mathematical formulas

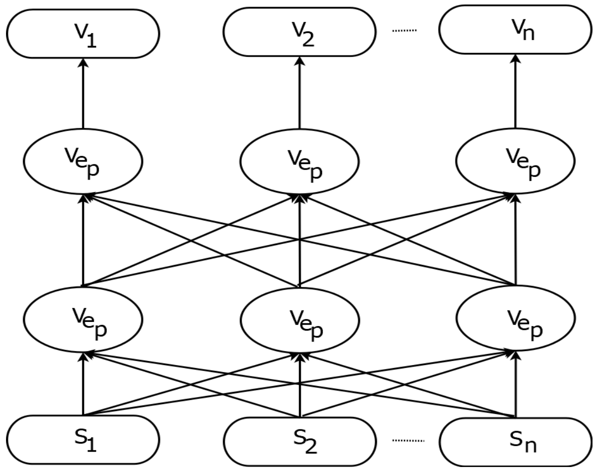

Figure 1. BERT network model

$$TPE\left(W_{loc,}2_{p+1}\right) = \cos\left(\frac{W_{loc}}{1000^{\frac{2_p}{d_{model}}}}\right). \quad (1)$$

$$TPE\left(W_{loc,}2_{p}\right) = \sin\left(\frac{W_{loc}}{1000^{\frac{2_p}{d_{model}}}}\right). \quad (2)$$

Where the word position in the text is W_{loc}, p symbolizes the dimension of input word window, and the dimension of encoding vector is d_{model}. The cosine function is the encoded representation of odd position. The sine function is the encoded representation of even position.

To collect more information at the word and sentence levels, the BERT language model simultaneously performs two tasks: Masking and Next Sentence Prediction. Masking is carried out by the Masked Language model which is identical to cloze filling (Lee et al., 2020). The objective of the Next Sentence Prediction tool is to understand the relationships between sentences and 50% correct sentence pairs are applied to train the model, tested with the remaining 50%.

Specifically, we use BERT to generate feature vectors from Text T1 and Text T2 together into a single token sequence ([CLS], t11, t12, ···, t1i, ···, t1m, [SEP], t21, t22, ···, t2i, ···, t2n, [SEP]), where [CLS] represents the special classification marker, [SEP] represents the special segment marker, t1i and t2i represent the ith token of the corresponding text.

Formally, the input is defined as X= (x1,x2,···,xi,···,xl), where xi∈R1×d is the embedding constructed by summing the ith token through the corresponding token, segment, and position embeddings, d is the maximum embedding dimension of the hidden layer, and l is the length of max input sequence. In layer j, a text embedding is denoted as E(j)= (e1,e2,···,ei,···,el), where ei∈R1×d coincides with the corresponding xi dimension.

2.1.5 Attention based bidirectional LSTM

RNN refers to a type of hidden-state recurrent feed-forward neural network. Classifying text is considered as a sequential modelling task. RNN has been frequently employed in various Natural Language Processing applications like sentiment analysis and text classification because of its sequential nature (Funahashi & Nakamura, 1993; Cao et al., 2017; Lee et al., 2019; Namee et al., 2023; Valentina & Songpan, 2023). Long short-term memory (LSTM) is an advanced RNN that can resolve vanishing gradient situations by memory cells (Schmidhuber & Hochreiter, 1997). Only the historical context can be accessed by the typical LSTM network but the absence of forward context can lead to an inadequate interpretation of the text's meaning. BiLSTM (Bidirectional LSTM) is a composition of forward and backward LSTM hidden layers. The context information is fully captured by incorporating these two hidden layers.

The proposed method presents a new technology by integrating BERT embedding, Bidirectional LSTM with Attention Mechanism.The proposed technology is designated as BERT-Attention-BiLSTM (BERT+ABiLSTM). In BERT+ABiLSTM, the BERT embedding layer captures semantic information from sentences. Then, we combine a forward and backward layer in order to obtain all the past and future context information.

The Attention Mechanism (AM) has been designed to allow the decoder to make use of the most important parts of the input sentences in a balanced manner, by combining all of the encoded input vectors into a weighted combination, with the most significant vectors treated as the highest weights. In ABiLSTM, two layers of attention learn the contextual information from preceding and following context. The preceding and following contextual information are combined in the AM which sends it in the Softmax function. The proposed BERT+ABiLSTM architecture is depicted in Figure 2.

The forward hidden layer ($\overrightarrow{hid_f}$) of BiLSTM symbolized as $\overrightarrow{LSTM}$which learns the information from LF_1 to LF_{100}. The backward hidden layer ($\overleftarrow{hid_b}$) of BiLSTM symbolized as $\overleftarrow{LSTM}$which learns the information from LF_{100} to LF_1. Finally, the outputs of ABiLSTM are formulated as below:

$$\overrightarrow{hid_f} = \overrightarrow{LSTM}(LF_n), n\varepsilon[1,100]. \tag{3}$$

$$\overleftarrow{hid_b} = \overleftarrow{LSTM}(LF_n), n\varepsilon[100,1]. \tag{4}$$

The forward annotation $\overrightarrow{hid_f}$ includes finding the input weights of forward annotation $\overrightarrow{Lu_f}$ by the single-layer perceptron. $\overrightarrow{Lu_f}$ is calculated as below:

$$\overrightarrow{Lu_f} = \tanh\left(w.\overrightarrow{hid_f} + b\right). \tag{5}$$

Where *b* and *w* are expressed accordingly as the bias and weight. The importance of the text is determined by the use of $\overrightarrow{Lu_f}$ and a context vector at the word level, $\overrightarrow{Lv_f}$. After that, Softmax is used to determine the normalised weight $\overrightarrow{att_f}$ of the text. The formulation is as follows:

$$\overrightarrow{att_f} = \frac{\exp\,(\overrightarrow{Lu_f} \times \overrightarrow{Lv_f})}{\sum_{i=1}^{M}(\exp\left(\overrightarrow{Lu_f} \times \overrightarrow{Lv_f}\right))}. \tag{6}$$

Where *M* is the number of words in the corpus and the exponential function is *exp(.)*. The contextual representation vector $\overrightarrow{Lv_f}$ at the word level can be viewed as a high-level representation over the words; throughout the training phase, it is generated at random and computed simultaneously.

Subsequently, the weighted sum of forward text annotations $\overrightarrow{att_f}$ and forward context information $\overrightarrow{hid_f}$ are used to find part of the attention layer outcomes, represented as F_c.

$$F_c = \sum(\overrightarrow{att_f} \times \overrightarrow{hid_f}). \tag{7}$$

And a weighted sum of backward text annotation $\overleftarrow{att_b}$ and backward context information $\overleftarrow{hid_b}$ are used to find part of the attention layer outcomes, represented as H_c.

$$H_c = \sum(\overleftarrow{att_b} \times \overleftarrow{hid_b}) \tag{8}$$

ABiLSTM collects annotations for a given sequence of features LF_n by putting together the forward contextual information F_c and the backward contextual information H_c.

Finally, we obtain the combined contextual representation$S = [F_c, H_c]$.The entire representations are decided to be the features of the text classification. In ABiLSTM, the probability distribution for classification is generated using the dropout and the Softmax layer. Dropout layers are used to avoid overfitting. Adam optimizer, which has been found to be an efficient and effective back propagation method, can be used to fine-tune model parameters (Kingma & Ba, 2014). Cross-entropy function reduces the chance of gradient disappearance in the stochastic gradient descent technique. To identify the loss L_{total}, the following function can be formed in equation 9.

$$L_{total} = -\frac{1}{NP}\sum_{Ss}[sl \ln r + (1 - sl)\ln(1 - r)] \tag{9}$$

Where NP is the total number of training data points, Ss is the sample size, sl is desired label of sample, and the result of ABiLSTM is r (predicted or actual label). Subsequently, the BiLSTM model predicts the class of the text (rumors) using Softmax classifier.

2.2 Methods

2.2.1 BERT+ABiLSTM language model

A deep learning ensemble model, BERT+ABiLSTM, was designed to identify COVID-19 related rumors. The general architecture of the BERT+ABiLSTM model is depicted in Figure 2.

The model is primarily consisted of three layers, BERT (used to generate vectors from words), BiLSTM (extracts semantic and temporal characteristics from the context) and Attention Layer (responsible for identifying words that are semantically connected to the context in order to aid the overall text understanding). The BERT+ABiLSTM model includes 256 LSTM units, and the feature vectors have the following sizes: Contextual representation

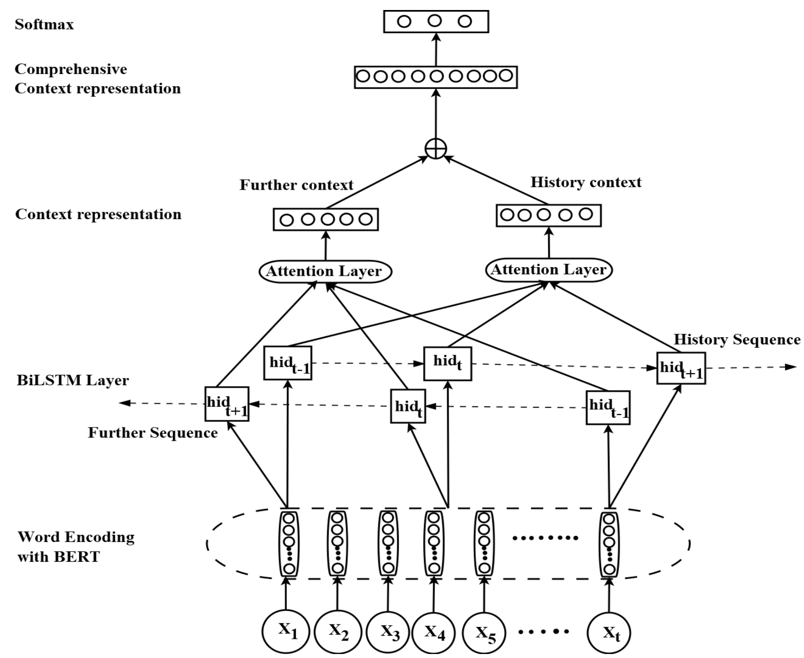

Figure 2. The architecture of the BERT+ABiLSTM model

of BERT: 768 dimensions (as in the base BERT model) and the output of the bidirectional LSTM layer: 256 dimensions per direction (forward and backward), together representing a total of 512 dimensions. Therefore, the total size of the feature vectors, including contextual representation, forward context information, and backward context information is: 768 + 512 = 1280 dimensions. In this work, the proposed model has been used to recognize rumors in the COVID-19 news and tweets. The particular procedures are as follows:

1. Consider rumor dataset as $R = \{r_1, r_2, r_3, \ldots\ldots\ldots., r_n\}$ where t^{th} rumor data is expressed as $r_i =< w_{i1}, w_{i2}, \ldots\ldots\ldots\ldots, w_{in} >$ and predefined class as $C = \{c_1, c_2, \ldots\ldots\ldots., c_n\}$.
2. Split the rumor dataset as training D_{Train} and testing D_{Test}.
3. The BERT+ABiLSTM classification model is trained on D_{Train} and tested on $D_{Test} = \{d_1, d_2, \ldots\ldots.., d_n\}$ and then calculates the Accuracy and F-measure according to Precision and Recall.

The key contributions of the BERT+ABiLSTM are listed below.

1. The BERT language model is used to generate vectors for dimension reduction. It extracts the semantic features of the text directly.
2. BiLSTM uses the word vectors and derives feature expressions including forward and backward contextual semantic information.
3. The attention mechanism is incorporated with the succeeding hidden layer and the preceding hidden layer of BiLSTM to design ABiLSTM, which helps to understand the vector interpretation of words in great detail.

Experiments are then performed to evaluate the efficiency of the suggested rumor classification technique on the dataset.

2.2.2 Dataset

The Rumor dataset is a list of news articles and tweets especially associated with the coronavirus. It consists of 4,129 articles of news and 2,705 tweets (Cheng et al., 2021).

There are three classes in the Rumor dataset. The class labeled “True” (sample size: 1878) indicates that the news is non-rumor and class labeled “False” (sample size: 3681) indicates that the news is a rumor. The third class labeled “Unverified” (sample size: 1275) indicates that the news cannot be validated at the time of collection. Table 1 illustrates the label definitions. The veracity of each class is depicted graphically in Figure 3. From Figure 3, it can be seen that the dataset is in the imbalanced form. Hence, it is essential to perform oversampling to balance the dataset for classification. After oversampling, we randomly split the total dataset (vocabulary size V is 90789 tokens) into 2 groups of 8:2. The ratio of 8:2 means 80% data for training and 20% for testing. Again, the training dataset is also split into 2 groups at a ratio of 8:2, for training and validation, respectively. In Table1, we have given some examples and explanation of the three classes from the rumor dataset. The text is True (T), when it is logic and explains the facts, e.g., “MERS is another strain of the Corona Virus”. The text is False (F), when it is imaginary or includes misleading information, e.g., “The current coronavirus has been manufactured in Wuhan”. Otherwise it is Unverified (U).

Table 1. Example of labels in the dataset

Term	Label	Examples and Description
Veracity	True (T)	The content is reasonable and provides facts, e.g., “MERS is another strain of the Corona Virus.”
	False (F)	The text is imaginary, or includes misleading facts, e.g., “The current coronavirus has been manufactured in Wuhan.”
	Unverified (U)	At the moment of labeling, it is difficult to determine the authenticity or truthfulness of the statement, e.g., “The study suggests that malaria drug can treat coronavirus.”

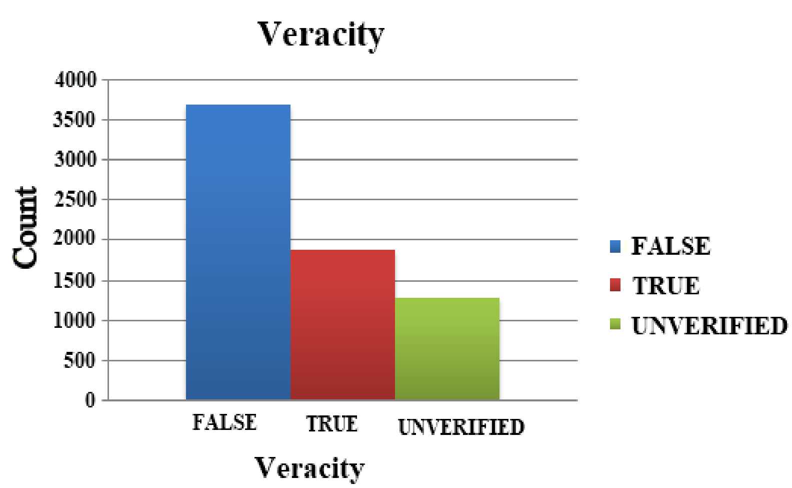

Figure 3. Number of veracity present in each class of the rumor dataset

2.2.3 Data preprocessing

Occasionally, data gathered from social media (such as Twitter) sources may contain some un-structured data elements. These un-structured data elements include hyperlinks, Twitter-specific terminology such '\#', '@', as well as single letter words, digits, etc. These can cause inconsistency in the classification. To improve classifier efficiency, the dataset must be cleaned. We perform the preprocessing task to eradicate un-structured data elements and perform basic NLP task like tokenization.

Imbalanced data frequently creates problem in the classification task. It happens when there are more data in one class than in any other classes (Akkaradamrongrat et al., 2019). In this situation, the data of minority class is insufficient for expression of concepts. The solution to this problem is data oversampling. Data oversampling is performed to create synthetic data from the original training samples.

The creation of synthetic data embedding vectors for correct representation of text is a challenging task. These vectors capture the semantics and contextual feature of the text accurately (Chen et al., 2014). Undersampling is another technique used to balance uneven datasets, which involves the reduction in size of the majority class to match the size of the minority class by randomly discarding samples from the majority class. While undersampling can be effective in some cases, it has its limitations, such as loss of ınformation (He et al., 2008), reduced model performance (Chawla et al., 2010), bias towards minority class (He & Garcia, 2009), and increased variability (Batista et al., 2004) and may not always be suitable for balancing text datasets (Mohammed & Abdullah, 2020). Hence, the data oversampling technique has been applied in this case.

SMOTE (synthetic minority over-sampling technique) is intended to be used with unbalanced datasets for learning (Chawla et al., 2003; Chawla et al., 2002). It involves interpolating between adjacent minority class samples to generate synthetic minority class samples. However, SMOTE suffers from the problem of over generalization (Han et al., 2005). DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a clustering technique based on density (Xu et al., 2019). It can identify clusters of various sizes and shapes from a huge quantity of data incorporating noise and boundaries. However, DBSCAN is not effective when dealing with the samples at the borderline (Ester et al., 1996). Hence, we have employed a density-based synthetic minority oversampling approach (DSMOTE) that combines the advantages of optimized DBSCAN and SMOTE. DSMOTE includes three phases as listed below.

Phase 1: Optimized DBSCAN is used to classify minority class samples into three categories, namely core samples, border line samples and noise samples. Then, minority class noise samples are eliminated.

Input: Original dataset *X*, distance threshold *eps=0.5*, density threshold *MinPts=4*, and minority samples X_{min}.

Output: Synthetic minority class samples.

Step 1: Select each unvisited point *x* in dataset *X*.

Step 2: *x* is considered as a core sample if it has more than *MinPts* points within *eps* andgenerates a new cluster for all core samples. All non-core samples which have fewer than *MinPts* within *eps* but are in the neighborhood of a core sample are identified as borderline samples. If *x* is neither a core sample nor a borderline sample, it is identifiedas a noise.

Step 3: Repeat step 2 for the remaining unvisited data points in *X*.

Step 4: We divide the undetermined borderline samples. Some borderline samples belong to the cluster of the core samples when all core samples are within their eps. Other borderline samples are having two core samples within its *eps*. In such situation, we compute the Euclidean distance between the sample at the boundary and the two core samples, and clustered it with the core sample which has the minimum distance.

Step 5: Samples which are not core samples or border samples get removed from the minority dataset and labeled as noise.

Suppose *Cs* denotes the core sample set and *B* denotes the borderline sample set, then $Cs = \{cs_1, cs_2, \dots \dots \dots \dots . . cs_k\}$, and $B = \{b_1, b_2, \dots \dots \dots . ., b_m\}$. Here, *k* represents the total number of examples in *Cs*, and *m* represents the total number of examples in *B*.

Phase 2: Oversample the core samples. For every cs_p $(p = 1,2, \dots \dots \dots \dots . ., k)$ in the minority core samples *Cs*, *N* samples are randomly selected from *eps* neighborhood. The difference $d_j (j = 1,2, \dots \dots \dots \dots . ., N)$ between cs_p and selected samples from *eps* (distance threshold) neighborhood are calculated and a random number r_j is selected (between 0 to 1). Then we get the oversamples s_j for the core points as formulated below.

$$s_j = cs_p + r_j \times d_j, \quad j = 1,2,3, \dots \dots \dots \dots . . N. \tag{10}$$

The above process is repeated for each sample cs_p in *Cs*, and finally $k \times N$ new samples are synthesized.

Phase 3: Oversample the borderline samples. We compute the cluster center of minority samples as

$$c_{center} = \left(\frac{1}{k} \sum_{p=1}^{k} cs_{p1}, \frac{1}{k} \sum_{p=1}^{k} cs_{p2}, \dots \dots \dots \frac{1}{k} \sum_{p=1}^{k} cs_{pr} \right) \tag{11}$$

Then for each sample b_i in *B*, we determine the difference d_j between b_i and c_{center} and select a random number r_j which is between 0 to 1. Then we get the oversamples s_j for the borderline points as formulated in the following equation

$$s_j = c_{center} + r_j \times d_j, \quad j = 1,2,3, \dots \dots \dots \dots . . N. \tag{12}$$

The above process is repeated for each sample b_i in *B*, and finally $m \times N$new samples can be synthesized.

3. Results and Discussion

We have applied the oversampling technique on the dataset before splitting and training with it. This oversampling technique is used to make sure that there are enough samples in the minority class for the proposed model, as shown in Figure 4 and Figure 5. Figure 4 represents the distribution of data points of tweets for each class before oversampling, and Figure 5 represents the distribution of data points of tweets for each class after oversampling. In these figures, it can be seen that after applying oversampling, the dataset is in the balanced form. The number of synthetic data point in the “True” class is 1803, and the number in the “Unverified” class is 2406 after employing the oversampling technique.

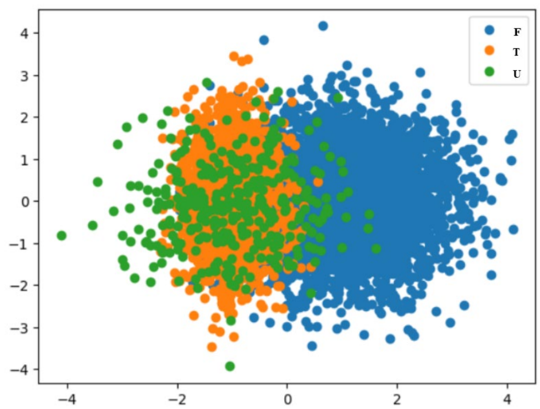

Figure 4. Distribution of data points before oversampling

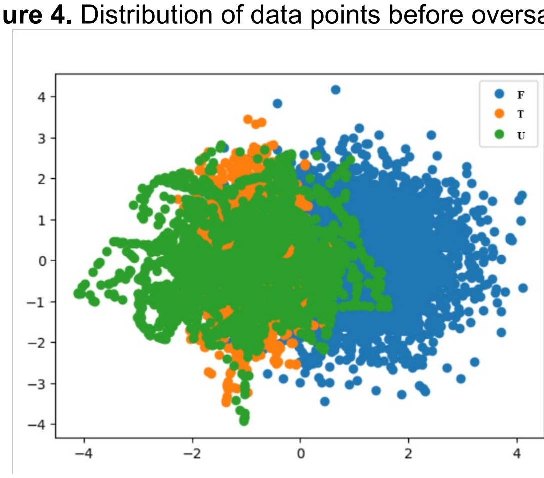

Figure 5. Distribution of data points after oversampling

For example, consider the following sentence for oversampling from the dataset, “remdesivir has only been approved for a clinical trial, not for consumer use”. After applying the oversampling technique, we get "approved remdesivir clinical trial not consumer used".

While going through the classification annotations given for the proposed model, we have observed that for most cases, the proposed model predicts the actual class (“True” or “False”) but there are also some cases where the proposed model fails to predict the actual class as discussed below.

For example, our proposed model predicts that “Remdesivir has only been approved for a clinical trials, not for consumer use” is true, but our model also believe that “Remdesivir has been approved for consumer use” is true. In addition, the model predicts that the label “If you get sick with coronavirus, Donald Trump can make you stay home” is false, and it seems that the models are correct. However, the model predicts that the label “If you get sick with coronavirus, a U.S. government official in your community can make you stay home” is also false. These examples demonstrate that in a few cases, the model misclassifies the label, since the model just outputs the labels “true”, “false”, or “unverified” based on incorrectly learnt rules.

The findings in Table 2 indicate that the system expects a statement to be a rumor, but this does not necessarily suggest that the system understands the meaning of the statement. For instance, the system believes the statement “Remdesivir has only been approved for a clinical trial, not for consumer use” is true, however the system does not understand the actual significance of this statement. Hence, the system falsely feels that the rumor “Remdesivir has been approved for consumer use” is also true. A single word can be the focus about 90% of a statement, which is clearly not fair. After looking at the dataset, we noticed that the problem is that these words are spread out in a very uneven way. For instance, “Trump” occurs mostly in the “False” statements but “Modi” appears almost exclusively in “True” statements.

Next, we have examined the performance of proposed system and compared it to other systems using the same dataset. To assess the suggested method, we have employed various essential metrics such as precision, recall, F-measure, accuracy and μF1 score, as can be seen in Table 3.

Table 2. Some case studies on the rumor dataset

Rumor	Label	Predicted Label
Remdesivir has only been approved for a clinical trial, not for consumer use	True	True
Remdesivir has been approved for consumer use	False	True
If you get sick with coronavirus, Donald Trump can make you stay home	False	False
If you get sick with coronavirus, a U.S. government official in your community can make you stay home	True	False
Coronavirus sends India into lockdown as trains halted, Modi says ‘please save yourself’	True	True
Prime Minister Narendra Modi has announced a week-long internet shut down in India	False	True
Donald Trump announced that Roche Diagnostics will launchCOVID-19 vaccine on Sunday	False	False
Trump Has Sabotaged America's Response to the Coronavirus Pandemic	True	False

From Table 3, the model proposed in this study obtained the precision, recall, F-measure, accuracy, and μF1score of 0.80, 0.81, 0.80, 80.71%, and 90.85, respectively, which are much better than other models in most of the cases such as BERT (precision: 0.63, recall: 0.69, f-measure: 0.66, accuracy: 65.73%, and μF1: 72.34), BERT+LSTM (precision: 0.68, recall: 0.74, f-measure: 0.71, accuracy: 69.52%, and μF1: 78.22), and BERT+LSTM with attention layer (precision: 0.63, recall: 071, f-measure: 0.67, accuracy 72.22%, and μF1: 82.40). Next, we demonstrate a comparison between our suggested system and the system (BERT + LSTM (VAE)) presented by Cheng et al. (2021) in Table 4.

Table 3. Comparative results of proposed system with other state-of-the-art systems

System Name	Precision	Recall	F-measure	Accuracy	µF1
BERT	0.63	0.69	0.66	65.73%	72.34
BERT + LSTM	0.68	0.74	0.71	69.52%	78.22
BERT + LSTM (attention)	0.63	0.71	0.67	72.22%	82.40
BERT + ABiLSTM [Proposed model]	0.80	0.81	0.80	80.71%	90.85

Table 4. Comparative result of proposed system with other existing systems

System Name	Parameter Selection	Learning Rate	Number of Parameters	µF1 Score
BERT + ABiLSTM (attention) [Proposed Model]	Embedding: 768 Batch: 32 Dropout: 0.5 Optimizer: Adam Stochastic Activation function: "Softmax" Balancing: DSMOTE (Over-sampling)	0.001	$\sim 123\text{M}$	90.85
BERT + LSTM (VAE) (Cheng et al., 2021)	Embedding: 512 Batch: 16 Dropout: 0.1 Optimizer: Adam Stochastic Activation function: "Softmax" Balancing: ADASYN (Over-sampling)	2×10^{-5}	$\sim 110\text{M}$	85.98

Table 4 shows that the proposed BERT+ABiLSTM system employs an embedding size of 768, while the BERT + LSTM (VAE) with ADASYN (Over-sampling) system employs an embedding size of 512. In the proposed system, the batch size used for training is 32, while BERT + LSTM (VAE) with ADASYN (Over-sampling) uses a batch size 16. We have used back propagation technique and Adam's stochastic optimizer to train the network over time. It has 123M different trainable parameters whereas BERT + LSTM (VAE) with ADASYN (Over-sampling) uses 110M trainable parameters. The µF1score for the proposed model is 90.85 whereas the µF1score for BERT + LSTM (VAE) with ADASYN (Over-sampling) is 85.98. Therefore, it can be observed from Table 4 that the proposed deep learning model significantly outperforms the other state-of-the-art models.

The evaluation of COVID-19 rumor classification can be obtained in terms of statistical measures namely the True Positives (TP), True Negatives (TN), False Positive (FP) and False Negatives (FN). The Confusion Matrix of the proposed BERT + ABiLSTM is shown in Table 5.

A further comparative evaluation of the proposed system and BERT + LSTM (VAE) with ADASYN (Over-sampling) regarding accuracy and loss function on the validation dataset are depicted in Figure 6 and Figure 7, respectively. From Figure 6, we have observed that the accuracy of the proposed system is significantly higher than the existing BERT + LSTM (VAE) with ADASYN model. In addition, Figure 7 indicates that the loss of the proposed system is less than the existing BERT + LSTM (VAE) with ADASYN model.

Table 5. Confusion matrix of the proposed BERT + ABiLSTM on test data

		Confusion Matrix	
		Actual Class	
		Positive	Negative
Predicted Class	Positive	5535 (TP)	1383 (FP)
	Negative	1298 (FN)	265 (TN)

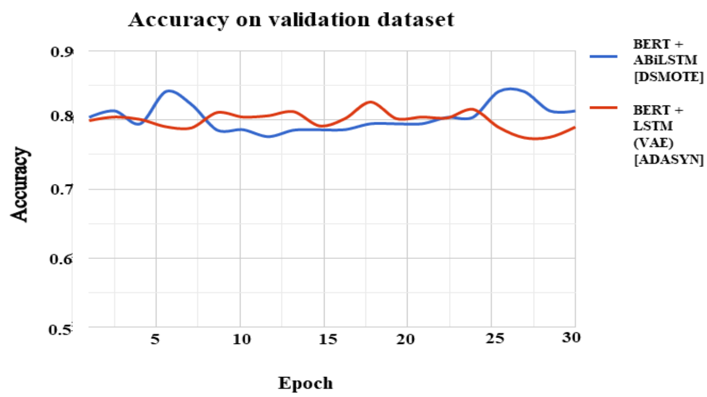

Figure 6. Comparison of the proposed model and BERT + LSTM (VAE) with ADASYN (Over-sampling) in terms of accuracy on validation dataset

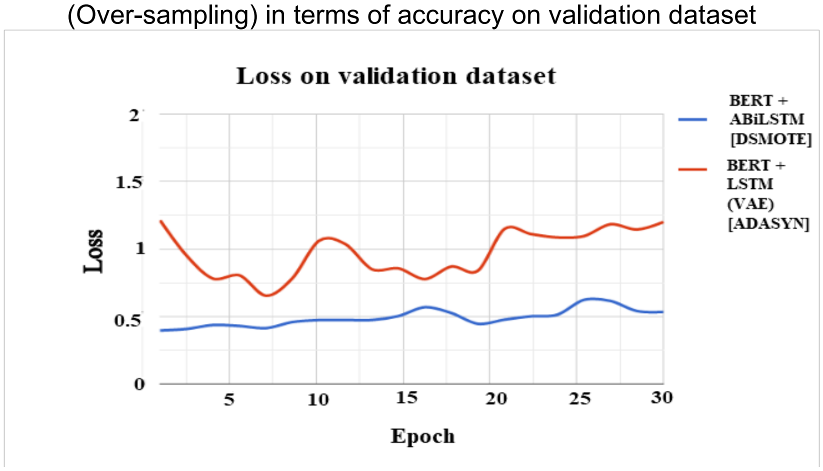

Figure 7. Comparison of the proposed model and BERT + LSTM (VAE) with ADASYN (Over-sampling) in terms of loss on validation dataset

4. Conclusions

This article introduces the BERT+ABiLSTM deep learning-based approach for COVID-19 rumor classification. The BERT word embedding layer, Attention Mechanism, and BiLSTM are combined together to improve the classifier's performance using different balanced candidate methods. By efficiently identifying and classifying the rumors into relevant classes, the developed model can help to avoid unnecessary panic-anxiety, can maintain socio-psychological and economic factors and can assist the government to deal with the

COVID-19 crisis by taking necessary precautionary measures. To examined the effectiveness of our suggested scheme, experiments on the COVID-19 rumor dataset were carried out. The comparative analysis demonstrates that the proposed technique is superior to existing state-of-the-art system techniques.

5. Conflicts of Interest

The author declares that they have no conflict of interest.

ORCID
Rakesh Dutta https://orcid.org/0000-0002-8590-4687
Mukta Majumder https://orcid.org/0000-0003-2608-5762

References

Alkhodair, S. A., Ding, S. H. H., Fung, B. C. M., & Liu, J. (2020). Detecting breaking news rumors of emerging topics in social media. *Information Processing and Management*, 57(2), Article 102018. https://doi.org/10.1016/j.ipm.2019.02.016

Aker, A., Sliwa, A., Dalvi, F. & Bontcheva, K. (2019). Rumour verification through recurring information and an inner-attention mechanism. *Online Social Networks and Media*, 13, Article 100045. https://doi.org/10.1016/j.osnem.2019.07.001

Akkaradamrongrat, S., Kachamas, P., & Sinthupinyo, S. (2019). Text generation for imbalanced text classification. In *Proceedings of the 16th international joint conference on computer science and software engineering* (pp. 181-186). IEEE. https://doi.org/10.1109/JCSSE.2019.8864181

Asghar, M. Z., Habib, A., Habib, A., Khan, A., Ali, R., & Khattak, A. (2021). Exploring deep neural networks for rumor detection. *Journal of Ambient Intelligence and Humanized Computing*, 12, 4315-4333. https://doi.org/10.1007/s12652-019-01527-4

Batista, G. E. A. P. A., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, 6(1), 20-29. https://doi.org/10.1145/1007730.1007735

Bourgonje, P., Schneider, J. M., & Rehm, G. (2017). From clickbait to fake news detection: an approach based on detecting the stance of headlines to articles. In *Proceedings of the 2017 EMNLP workshop: Natural language processing meets journalism* (pp. 84-89). Association for Computational Linguistics. https://doi.org/10.18653/v1/W17-4215

Bratu, S. (2020). The fake news sociology of COVID-19 pandemic fear: Dangerously inaccurate beliefs, emotional contagion, and conspiracy ideation. *Linguistic and Philosophical Investigations*, 19, 128-134. https://doi.org/10.22381/lpi19202010

Cao, W., Song, A., & Hu, J. (2017). Stacked residual recurrent neural network with word weight for text classification. *IAENG International Journal of Computer Science*, 44(3), 277-284.

Chawla, N. V. (2010). Data mining for imbalanced datasets: An overview. In M. Oded & R. Lior (Eds). Data mining and knowledge discovery handbook (2nd ed., pp. 875-886). Springer. https://doi.org/10.1007/978-0-387-09823-4_45

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16, 321-357. https://doi.org/10.1613/jair.953

Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W. (2003). SMOTEBoost: Improving prediction of the minority class in boosting. In *Proceedings of the 7th European* *conference on principles and practice of knowledge discovery in databases* (pp. 107-119). Springer. https://doi.org/10.1007/978-3-540-39804-2_12

Chen, T., Xu, R., Liu, B., Lu, Q., & Xu, J. (2014). *WEMOTE-Word embedding based minority oversampling technique for imbalanced emotion and sentiment classification*. https://sentic.net/wisdom2014chen.pdf.

Cheng, M., Wang, S., Yan, X., Yang, T., Wang, W., Huang, Z., Xiao, X., Nazarian, S., & Bogdan, P. (2021). A COVID-19 rumor dataset. *Frontiers in Psychology*, 12, Article 644801. https://doi.org/10.3389/fpsyg.2021.644801

DiFonzo, N., & Bordia, P. (2007). *Rumor psychology: Social and organizational approaches*. Washington: American Psychological Association. https://doi.org/10.1037/11503-000

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd International conference on knowledge discovery and data mining* (pp. 226-231). Association for the Advancement of Artificial Intelligence.

Funahashi, K.-I., & Nakamura, Y. (1993). Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6(6), 801-806. https://doi.org/10.1016/S0893-6080(05)80125-X

Guo, C., Lin, S., Huang, Z., & Yao, Y. (2022). Analysis of sentiment changes in online messages of depression patients before and during the COVID-19 epidemic based on BERT+ BiLSTM. *Health Information Science and Systems*, 10(1), Article 15. https://doi.org/10.1007/s13755-022-00184-w

Guo, H., Cao, J., Zhang, Y., Guo, J., & Li, J. (2018). Rumor detection with hierarchical social attention network. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 943-951). ACM. https://doi.org/10.1145/3269206.3271709

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning. In *Proceedings of the international conference on intelligent computing* (pp. 878-887). Springer. https://doi.org/10.1007/11538059_91

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284. https://doi.org/10.1109/TKDE.2008.239

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *Proceedings of the international joint conference on neural networks* (pp. 1322-1328). IEEE. https://doi.org/10.1109/IJCNN.2008.4633969

Hui, H., Zhou, C., Lü, X., & Li, J. (2020). Spread mechanism and control strategy of social network rumors under the influence of COVID-19. *Nonlinear Dynamics*, 101(3), 1933-1949. https://doi.org/10.1007/s11071-020-05842-w

Jin, Z., Cao, J., Guo, H., Zhang, Y., & Luo, J. (2017). Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In *Proceedings of the 25th ACM international conference on multimedia* (pp. 795-816). ACM. https://doi.org/10.1145/3123266.3123454

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv:1412.6980*, https://doi.org/10.48550/arXiv.1412.6980

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., & Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4), 1234-1240. https://doi.org/10.1093/bioinformatics/btz682

Lee, Y. H., Kim, D. W., & Lim, M. T. (2019). A two-level recurrent neural network language model based on the continuous Bag-of-Words model for sentence classification. *International Journal on Artificial Intelligence Tools*, 28(1), Article 19500027. https://doi.org/10.1142/S0218213019500027

Li, F., Jin, Y., Liu, W., Rawat, B. P. S., Cai, P., & Yu, H. (2019). Fine-tuning bidirectional encoder representations from transformers (BERT)–based models on large-scale electronic health record notes: an empirical study. *JMIR Medical Informatics*, 7(3), Article e14830. https://doi.org/10.2196/14830

Long, Y., Lu, Q., Xiang, R., Li, M., & Huang, C.R. (2017). Fake news detection through multi-perspective speaker profiles. In *Proceedings of the 8th international joint conference on natural language processing* (pp. 252-256). Asian Federation of Natural Language Processing.

Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B. J., Wong, K. F., & Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks. In *Proceedings of the 25th international joint conference on artificial intelligence* (pp. 3818-3824). AAAI Press.

Mihaylov, T., & Nakov, P. (2016). Hunting for troll comments in news community forums. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics* (pp. 399-405). Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-2065

Mihaylov, T., Georgiev, G., & Nakov, P. (2015a). Finding opinion manipulation trolls in news community forums. In *Proceedings of the 19th conference on computational natural language learning* (pp. 310-314). Association for Computational Linguistics. https://doi.org/10.18653/v1/K15-1032.

Mihaylov, T., Koychev, I., Georgiev, G., & Nakov, P. (2015b). Exposing paid opinion manipulation trolls. In *Proceedings of the international conference recent advances in natural language processing* (pp. 443-450). INCOMA Ltd.

Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques: overview study and experimental results. In *Proceedings of the 11th international conference on information and communication systems* (pp. 243-248). IEEE. https://doi.org/10.1109/ICICS49469.2020.239556

Namee, K., Polpinij, J., & Luaphol, B. (2023). A hybrid approach for aspect-based sentiment analysis: A case study of hotel reviews. *Current Applied Science and Technology*, 23(2), https://doi.org/10.55003/cast.2022.02.23.008

Nguyen, T. N., Li, C., & Niederée, C. (2017). On early-stage debunking rumors on twitter: Leveraging the wisdom of weak learners. In *Proceedings of the 9th international conference on social informatics* (pp. 141-158). Springer. https://doi.org/10.1007/978-3-319-67256-4_13

Pham, T. T. (2018). *A Study on Deep Learning for Fake News Detection.* [Master's Thesis, Japan Advanced Institute of Science and Technology]. JAIST. https://dspace.jaist.ac.jp/dspace/bitstream/10119/15196/3/paper.pdf

Qazvinian, V., Rosengren, E., Radev, D., & Mei, Q. (2011). Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the 2011 conference on empirical methods in natural language processing* (pp. 1589-1599). Association for Computational Linguistics.

Sahni, H., & Sharma, H. (2020). Role of social media during the COVID-19 pandemic: Beneficial, destructive, or reconstructive? *International Journal of Academic Medicine*, 6(2), 70-75. https://doi.org/10.4103/IJAM.IJAM_50_20

Schmidhuber, J., & Hochreiter, S. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735-1780.

Tripathy, R. M., Bagchi, A., & Mehta, S. (2010). A study of rumor control strategies on social networks. In *Proceedings of the 19th ACM international conference on information and knowledge management* (pp. 1817-1820). Association for Computing Machinery. https://doi.org/10.1145/1871437.1871737

Valentina, S., & Songpan, W. (2023). Features extraction based on probability weighting for fake news classification on social media. *Current Applied Science and Technology*, 23(2), 1-18. https://doi.org/10.55003/cast.2022.02.23.014

Vijeev, A., Mahapatra, A., Shyamkrishna, A., & Murthy, S. (2018). A hybrid approach to rumour detection in microblogging platforms. In *Proceedings of the international conference on advances in computing, communications and informatics* (pp. 337-342). IEEE. https://doi.org/10.1109/ICACCI.2018.8554371

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, 359(6380), 1146-1151. https://doi.org/10.1126/science.aap9559

Wu, K., Yang, S., & Zhu, K. Q. (2015). False rumors detection on Sina Weibo by propagation structures. In *Proceedings of the 31st international conference on data engineering* (pp. 651-662). IEEE.

Xu, X., Chen, W., & Sun, Y. (2019). Over-sampling algorithm for imbalanced data classification. *Journal of Systems Engineering and Electronics*, 30(6), 1182-1191. https://doi.org/10.21629/JSEE.2019.06.12

Yang, F., Liu, Y., Yu, X., & Yang, M. (2012). Automatic detection of rumor on sina weibo. In *Proceedings of the ACM SIGKDD workshop on mining data semantics* (pp. 1-7). Association for Computing Machinery. https://doi.org/10.1145/2350190.2350203

Yang, J., & Pan, Y. (2021). COVID-19 rumor detection on social networks based on content information and user response. *Frontiers in Physics*, 9, Article 763081. https://doi.org/10.3389/fphy.2021.763081

Zhao, Z., Resnick, P., & Mei, Q. (2015). Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th international conference on world wide web* (pp. 1395-1405). The Web Conference. https://doi.org/10.1145/2736277.2741637

Zubiaga, A., Liakata, M., & Procter, R. (2017). Exploiting context for rumour detection in social media. In *Proceedings of the 9th international conference on social informatics* (pp. 109-123). Springer. https://doi.org/10.1007/978-3-319-67217-5_8.