

Ensemble approach to rumor detection with BERT, GPT, and POS features

Barsha Pattanaik¹, Sourav Mandal¹, Rudra Mohan Tripathy¹, Arif Ahmed Sekh²

¹School of Computer Science and Engineering, XIM University, Bhubaneswar, India

²Department of Computer Science, UiT The Arctic University of Norway, Troms, Norway

Article Info

Article history:

Received Jun 26, 2024

Revised Oct 7, 2024

Accepted Nov 19, 2024

Keywords:

BERT

BiLSTM

Ensemble model

GPT

Part of speech

Rumor detection

ABSTRACT

As vast amounts of rumor content are transmitted on social media, it is very challenging to detect them. This study explores an ensemble approach to rumor detection in social media messages, leveraging the strengths of advanced natural language processing (NLP) models. Specifically, we implemented three distinct models: (i) generative pre-trained transformer (GPT) combined with a bidirectional long short-term memory (BiLSTM) network; (ii) a model integrating part-of-speech (POS) tagging with bidirectional encoder representations from transformers (BERT) and BiLSTM, and (iii) a model that merges POS tagging with GPT and BiLSTM. We included additional features from the dataset in all these models. Each model captures different linguistic, syntactical, and contextual features within the text, contributing uniquely to the classification task. To enhance the robustness and accuracy of our predictions, we employed an ensemble method using hard voting. This technique aggregates the predictions from each model, determining the final classification based on the majority vote. Our experimental results demonstrate that the ensemble approach significantly outperforms individual models, achieving superior accuracy in identifying rumors. To determine the performance of our model, we used PHEME and Weibo datasets available publicly. We found our model gave 97.6% and 98.4% accuracy, respectively, on the datasets and has outperformed the state-of-the-art models.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Authors:

Barsha Pattanaik

School of Computer Science and Engineering, XIM University

Bhubaneswar-752050, Odisha, India

Email: barsha@xustudent.edu.in

1. INTRODUCTION

In this digital age, social media platforms have become ubiquitous, serving as primary channels for information dissemination and communication. While these platforms offer numerous benefits, they also facilitate the rapid spread of misinformation and rumors, which can have significant societal impacts. The proliferation of false information on social media can lead to public panic, misinformation crises, and harm to individuals or groups. Therefore, developing effective techniques for detecting and mitigating the spread of rumors has become a critical area of research. Traditional rumor detection methods often rely on manual verification, which is time-consuming and impractical given the vast volume of data generated on social media. Automated rumor detection systems, leveraging advancements in natural language processing (NLP), machine learning (ML), and deep learning (DL), offer a promising solution to this challenge. Recent developments in DL-based models, such as generative pre-trained transformer (GPT) and bidirectional encoder representa-

tions from transformers (BERT), have demonstrated remarkable capabilities in understanding and generating human-like text, making them suitable for tackling complex linguistic tasks.

Previous research focused on the role of textual characteristics in rumor identification, although it has not specifically considered the effect of additional social media factors, such as the number of retweets, user follows, and user friends. These features offer an essential context for comprehending the dissemination and influence of rumors. Previous studies have predominantly concentrated on language-centric models, ignoring the possibilities of social media information. This study addresses this gap by integrating text embeddings and supplementary social media data into an ensemble model, providing a more comprehensive approach to rumor identification. Furthermore, adding to the novelty, we have incorporated both BERT and GPT embeddings to study the semantic understanding and syntactic features along with contextual embeddings. In this study, we propose an ensemble-based approach for rumor detection, integrating the strengths of multiple NLP models. For this purpose, we extracted four different types of features using NLP tools and libraries. All the features are explained below, along with an overview of our proposed methodology.

Features extraction and methodology includes:

- We use pre-trained BERT embeddings to generate dense vector embeddings that capture the semantic meaning and context of the words in the messages.
- GPT embeddings, like BERT embeddings, capture the contextual meaning of a text but are unidirectional, considering context from left to right. They help the model understand language features and semantics.
- Part of speech (POS) tag features provide syntactic information crucial for understanding sentence structure and identifying patterns in rumors. Each word is tagged with its part of speech (noun and verb), and these tags are converted into one-hot encoded vectors.
- Additional features (AF) include `retweet_count`, `'user.followers_count'`, `'user.friends_count'`, and `'favorite_count'`, normalized using a standard scaler. These features quantify post engagement and user influence, indicating the spread and credibility of the message.

These features include numerical metrics associated with the social media post, such as the number of favorites, retweets, and user-specific information like follower and friend counts to provide quantitative information about the post's engagement and the user's social influence, which can be indicative of the spread and credibility of the message. By using these features, we developed three distinct models as discussed in section 3.2. To enhance the overall performance, we employed an ensemble method of these three models using hard voting, wherein the majority vote from the individual models determines the final classification. This ensemble approach aims to leverage the diverse strengths of each model, resulting in improved accuracy and reliability in detecting rumors. We have used two publicly available rumor datasets-PHEME [1], and Weibo [2] for our experiments in this study. We have translated the Weibo dataset into English language using Google Translator discussed in section 3.1. Therefore, we refer to this dataset as 'WeiboE'.

The following are our significant contributions in this study:

- i) We have used multiple baseline architectures and pre-trained models, such as BERT and GPT, to create alternative neural networks for rumor detection.
- ii) We have presented an ensemble classifier for rumor detection (the first of its kind), including thorough research and performance analysis and improving the baseline.
- iii) Our proposed model outperformed all the previous rumor detection systems on both the standard datasets-PHEME and Weibo.

Next, we describe related work in section 2, the proposed methodology in section 3, result analysis in section 4, and conclude in section 5.

2. RELATED WORK

In literature, numerous techniques have been developed to detect fake news [3]–[5]. Recently, the research community has also focused on identifying rumors. Although fake news and rumors are distinct, the methods employed by researchers are quite similar, involving text or document classification using NLP and advanced ML or DL techniques. With the increasing prevalence of rumor content on social media, researchers have devised various DL-based models to tackle this issue. Detailed information on these methods and their performance can be found in several survey papers [6]–[11]. In this section, we summarize some of the notable research that utilized ML or DL-based approaches and demonstrated strong performance on their respective datasets.

2.1. Traditional methods for rumor detection

Early approaches to rumor detection primarily relied on traditional ML techniques combined with handcrafted features. These methods typically involved two things: feature extraction by using lexical cues, metadata (e.g., user information, message propagation patterns), and temporal patterns, which were manually extracted. Second, ML algorithms like support vector machines (SVM), decision trees, and Naive Bayes were used to classify messages as rumors or non-rumors. For instance, Castillo *et al.* [12] discussed the relevance and significance of information quality particularly credibility of the information in the context of Twitter which is one of the fastest-growing social media platforms posting information both true and false rumors. Hence, they developed a method that applies SVM with factors like the number of retweets, data URLs, and credibility scores of the user publishing on the Twitter platform to filter out fake news. This approach offered improvement but it had its downsides since the features had to be extracted tediously by hand and might not always apply to other datasets or scenarios. Depending on the accuracy level of their study, which was about 86%, they managed to find answers to questions.

2.2. Use of deep learning models

New generations of the model were developed by many researchers as DL came in, which is capable of extracting features from raw text without requiring human manual intervention. Duration-based recurrent neural networks (RNNs), particularly long short-term memory (LSTM), and bidirectional LSTM (BiLSTM) networks [13], emerged as a rich source of capturing the sequential structure of textual data. Meanwhile, Ma *et al.* [2] further proposed a novel model adopting LSTM networks, which incorporated temporal information of tweet sequences to enhance the identification of existing rumor tweets. The accuracy on the Twitter dataset is 88.1% while on the WeiboE dataset is 91%. Using both datasets, Ruchansky *et al.* [14] proposed addressing fake news detection by using CSI (capture, score, integrate) model, based on RNNs and user and comments' features. RNN and convolutional neural network (CNN) [15] are used for capturing both temporal and content features hence giving high accuracy while the incorporation of user behavior significantly enhances the robustness of the model. There was approximately 89% accuracy when applied in the Twitter dataset and 95.3% in the Weibo dataset depending on the CSI model.

2.3. Use of transformer-based models

Later, deeper models like BERT, GPT and many others have transformed NLP by providing better encoding techniques of context information. These models do store deep semantic and syntactic features and are generally effective for the text classification problem. The paper by Devlin *et al.* [16] presented BERT, a totally new model that fine-tuned preemptively on the large text associated with the vocabulary obtained by Web Scraping and achieved state-of-the-art accuracy on numerous NLP enterprises through employing bidirectional context. This is mainly due to one of the considerable advantages of BERT that allows it to recognize the word context in both directions which is especially beneficial for recognizing the language used in rumors. Anggrainingsih *et al.* [17] developed a BERT-based approach for rumor detection by using sentence embedding to capture the contextual meanings of the message. GPT-2 was designed by Radford *et al.* [18], and its excellent capacity for language generation established through 'guessing' the next word in a given text results from capturing contextual relation dependencies. This work is one of the Paragon works in this group demonstrating that massive language models can perform a number of tasks without problem-specific training. Liu *et al.* [19] used different large language models such as GPT and BERT to check whether these models can detect rumors in social media by using both news and comments and their propagation information.

2.4. Use of ensemble methods

The combination of models has been referred to as ensemble methods due to the ability of the higher and numerous models to improve on the basic models. They can enable improvement by using the strengths of each model when it comes to improve the overall performance. Hard voting [20] works by taking a majority vote for categories while soft voting takes the probability that each model has assigned to categories. As can be seen from the above equations, both techniques have made enhancements in enhancing classification tasks by overcoming the demerits of separate models. Kotteti *et al.* [21] employed an ensemble of dL-based models by using CNNs, RNNs, and LSTMs for processing time-series data to improve the accuracy and robustness of rumor detection. They used features from the time-series data, including tweet content, user metadata, and network propagation patterns. The extracted features were then fused to create a comprehensive representation used for classification. The model gave 64.3%, which is 79% more in terms of micro F1-score on PHEME

datasets compared to the baselines. Recently, Yuan *et al.* [22] developed an ensembling model by using two different features, such as image and text. For text data, the authors used BERT and BiLSTM, gated recurrent unit (GRU) [23], and for image data, they used CNN. After that, they ensembled different models and used soft voting for the final classification. Nithya *et al.* [24] developed a hybrid model leverages the strengths of multiple advanced techniques in NLP and DL to effectively classify rumor texts, combining deep contextual understanding (using BERT), hierarchical feature extraction, feature importance analysis, and sequence modeling (using Bi-LSTM) into a cohesive framework.

Our work focused on developing three different models to capture different aspects of textual data, leveraging both semantic understanding and syntactic features along with contextual embeddings. Finally, an ensemble model is proposed to integrate the predictions from these models, enhancing the robustness and accuracy of the classification. We are the first to propose a model by considering different features and integrating them into an ensemble model.

3. PROPOSED METHOD

The objective of this research is to detect if a message or post is a rumor or not. We have considered a binary classification approach. For this research as a baseline model, we have used BERT embedding with the BiLSTM network [25], [26] developed for sentiment analysis and entity recognition for clinical tests, followed by a dense layer and softmax for classification. The BiLSTM processes the sequence of embeddings in both forward and backward directions, capturing dependencies from the past and future contexts.

3.1. Data collection and pre-processing

For our experiment, we use two publicly available datasets, such as PHEME [1] and Weibo [2] focusing on messages labeled as either rumors or non-rumors. The dataset was curated to ensure a balanced representation of both classes, allowing for effective training and evaluation of our models. The PHEME dataset (5,802 samples) provides a detailed breakdown of rumor (1,972 instances) and non-rumor (3,830 instances) tweets across several five events (OT: Ottawa shooting, GC: Germanwings crash, FE: Ferguson, CH: Charlihebbdo, SY: Sydneyside). Similarly, the Weibo dataset contains 2,313 rumors and 2,351 non-rumors. The Weibo dataset contains many features, but we only used four features such as ‘retweet_count’, ‘user.followers_count’, ‘user.friends_count’, and ‘favorite_count’ for our model. We translate the Weibo dataset originally available in the Chinese language to English using the “Google Translator” of ‘deep_translator’ package. The name of the dataset is given as WeiboE and the link to the dataset is available on the github [27] for future research. Weibo is not segregated in events like PHEME. The details about the datasets are explained below in a stacked bar graph in Figure 1 and Table 1 shows the sample messages of PHEME datasets containing AF and labels.

Data pre-processing involved several steps to clean and prepare the text data for model training. All the text breaks down into individual tokens (words or subwords). Then, common stop-words are removed that don’t have significant contributions to the meaning of a sentence. Then, lemmatization is done to make a word into its base forms. Annotation uses POS tags to capture the syntactic information. After tagging, we use padding and truncation to ensure uniform input lengths for batch processing. For example, in the case of the Ottawa shooting event, a sample of data is “Ottawa police are confirming a shooting at the War Memorial. Minutes ago. No other info. #cbcOTT #OTTnews” first tokenization is done. Subsequently, lowercasing transforms all tokens to lowercase to ensure uniformity. Stopwords like “are” and “at” are eliminated to emphasize more significant keywords. Punctuation and special characters, including hashtags, are removed to sanitize the data. The cleaned tokens obtained are [‘Ottawa’, ‘police’, ‘confirming’, ‘shooting’, ‘war’, ‘memorial’, ‘minutes’, ‘ago’, ‘info’, ‘cbcott’, ‘ottnews’]. Moreover, POS tagging can be utilized to provide grammatical classifications to each token, so offering enhanced linguistic context.

3.2. Experimental models

3.2.1. Baseline model- BERT and BiLSTM

We use a simple BERT with BiLSTM network (BERT+BiLSTM) developed by [25], [26] as the base model for our rumor detection task. We used the PHEME dataset of five events containing social media posts as input messages. The model used a pre-trained transformer, BERT, to extract contextual information for word embedding. Next, the embedding vectors from the BERT are sequentially fed to the BiLSTM network for learning bi-directional long-term dependencies of the words (vectors) across the input sentences. The concatenated and flattened vector for each sequence is then fed to the dense layer, followed by the softmax layer

for classification. We introduce three new variants of this baseline model, each using different feature combinations but sharing a common BiLSTM architecture which are explained in the sections below.

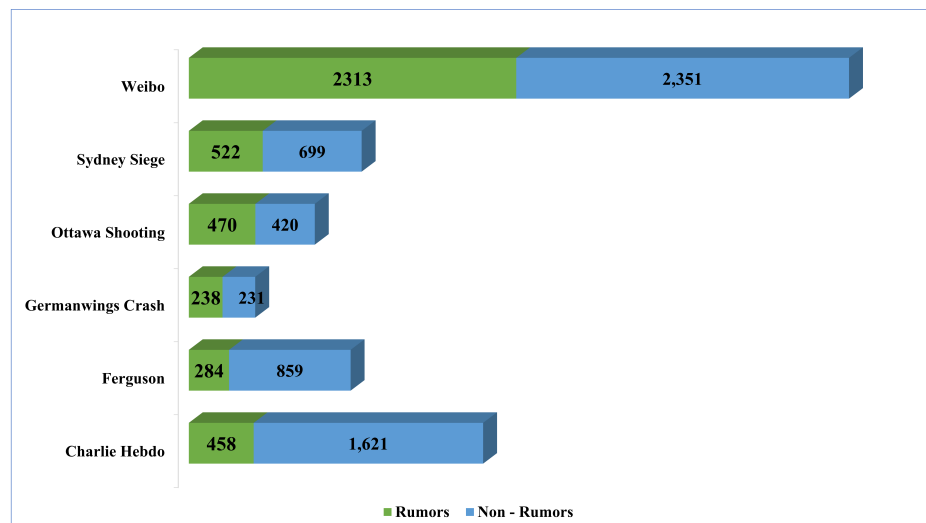


Figure 1. Rumors and non-rumors data distribution across PHEME and WeiboE datasets

Table 1. Sample data of PHEME and WeiboE datasets showing features with labels (0-non rumor, 1-rumor)

Sample texts	favorite_count	retweet_count	user.followers_count	user.friends_count	Label
Being black in this country is dangerous business. #Ferguson (PHEME)	117	198	25565	1593	0
Rest in Peace, Cpl. Nathan Cirillo. Killed today in #OttawaShooting http://t.co/YzLXYX5JJt http://t.co/8F0qAcj9sg (PHEME)	96	112	14793	1052	1
At 8:26 am on February 26, Li Tianyi was released on bail and is now returning home.(WeiboE)	33	1272	977	499	1
We are all grandsons, why are our realms so different? (WeiboE)	4	1073	32914	2180	0

3.2.2. Model-1 (GPT, AF, and BiLSTM)

In this model, we use two features: GPT embeddings and additional features, or AF as shown in Figure 2. We utilized the pre-trained GPT instead of BERT to generate contextualized embeddings for each word in the input message. GPT's capacity to understand and generate coherent text was leveraged to capture the nuanced context within the data. The embedding vectors from GPT, along with the other features, are then fed into a BiLSTM network (GPT+AF+BiLSTM) and the rest are the same as in the baseline model.

3.2.3. Model-2 (POS, BERT, AF, and BiLSTM)

Model 2 in Figure 3 uses three features, such as AF derived from the text, POS features, and BERT embeddings (POS+BERT+AF+BiLSTM). POS is a critical linguistic processing step that involves annotating each word in a sentence with its corresponding part of speech, such as noun, verb, and adjective. In the context of rumor detection, POS tag-based features serve several important purposes. Each token is annotated with its POS tag, providing additional syntactic information. These features can be particularly useful for capturing syntactic and grammatical nuances that purely word-based embedding techniques might miss. This enriched feature set can improve the overall performance of the models in detecting subtle cues indicative of rumors. In this model, the tokens and their POS are input into the BERT model to obtain rich, contextualized embedding vectors. Similar to the BERT+BiLSTM model, these embedding vectors are then processed through a BiLSTM network to capture sequential dependencies.

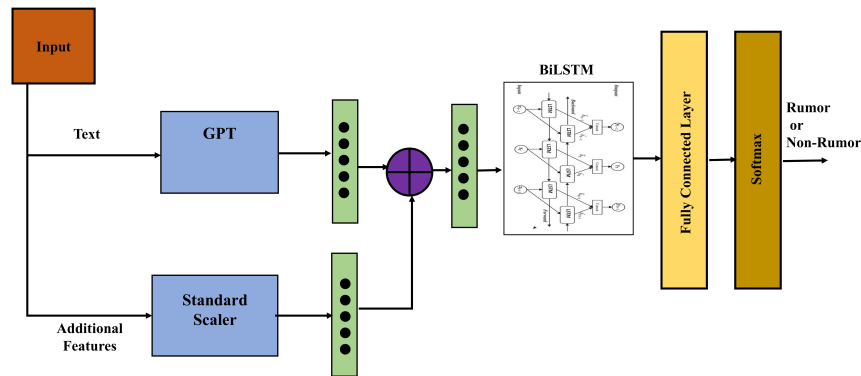


Figure 2. Model-1 with GPT with additional features and BiLSTM

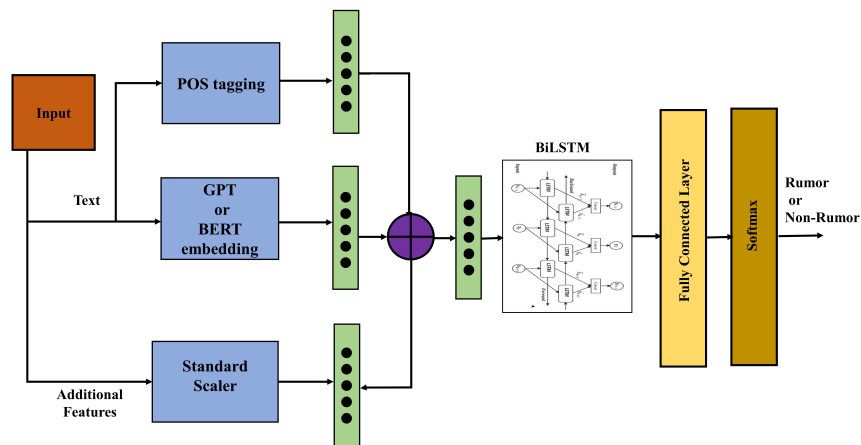


Figure 3. Model-2 and model-3; BERT and GPT interchangeably used to make model-2 or model-3

3.2.4. Model-3 (POS, GPT, AF, and BiLSTM)

Like the previous model, the model uses GPT embeddings instead of BERT (Figure 3) along with the other two features (POS+GPT+AF+BiLSTM). POS tagging boosts rumor detection by enhancing syntactic, contextual, and semantic understanding and enriching features for deep learning. The POS-tagged tokens are processed through the GPT model to generate embeddings. The embeddings are input into a BiLSTM network to capture contextual dependencies.

3.2.5. Detail procedure and descriptions

As described in the previous sections, we use four types of features as input to the BiLSTM. First, the tokenized and cleaned text is processed with NLTK's `pos_tag` function to obtain POS tags, assigning grammatical categories like nouns, verbs, and adjectives to each word. Each tokenized word is labeled with a POS tag using NLTK's `pos_tag` in a bid to extract the POS features. After that, these POS tags were encoded by a numerical vector using one-hot encoding. When included with models such as BERT and GPT, POS tags allow for science-based definitions of rumors that can capture the linguistic patterns of the phenomenon more accurately. When rumors make statements about a certain topic, they tend to do it with the use of certain adjectives or adverbs that inflate the fact at hand. The algorithms can learn these patterns more efficiently and enhance the accuracy of rumor detection when these POS tags are incorporated. In this step, independently we use one of the BERT or GPT tokenizers from the hugging face transformers [28] library to tokenize each text. Contextual embeddings are then obtained by passing the tokenized text with BERT or GPT. The additional features or AF are also floored and ceilinged before being scaled using the 'Standard Scaler' in 'scikit-learn'. The BERT or GPT embedding vectors, one-hot encoded POS features, and scaled additional features are combined into a single feature vector for each text using 'np.hstack' specific to the model's requirements, resulting in a comprehensive feature vector for each input text. The BiLSTM [29] model takes these combined feature vectors as

the input, processes them, and feeds them into the fully connected layer. The fully connected layer, followed by the softmax layer, is responsible for mapping the combined BiLSTM output to the class scores.

3.3. Model-4: proposed ensemble method

Figure 4 depicts our proposed ensemble model for rumor detection. We employed a hard voting mechanism [20] to combine the predictions from the three models discussed in the previous subsection. Each model independently classifies a message as a rumor or non-rumor, and the final classification is determined by the majority vote among the three models. This approach leverages the strengths and compensates for the weaknesses of individual models, aiming to enhance overall accuracy. Algorithm 1 explains the algorithm of our ensemble model with hard voting. Detail procedure of each model is described in section 3.2.5.

Through the utilization of these three models in an ensemble, we want to capitalize on:

- The capability of GPT to effectively capture long-range dependencies and contextual continuity.
- BERT’s bidirectional contextual comprehension augmented by POS tagging.
- Additional (supplementary) features (AF) to record behavioral indicators, including tweet and retweet frequencies.
- BiLSTM’s sequential modeling facilitates the capturing of both forward and backward text dependencies.

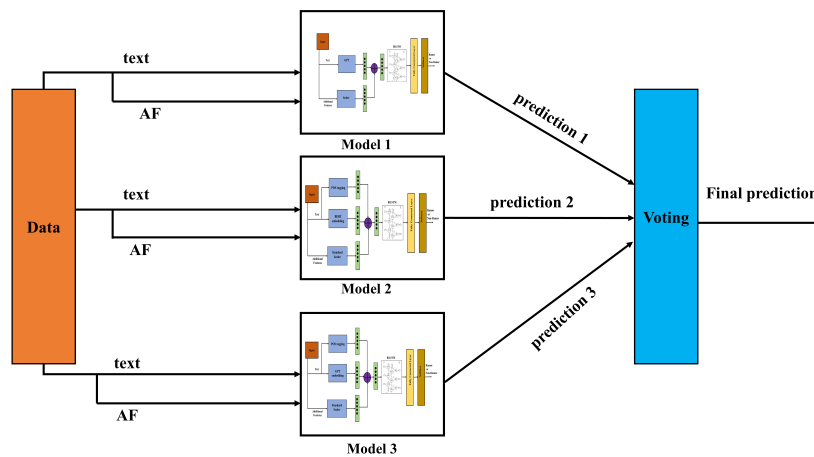


Figure 4. Proposed ensemble model for rumor detection

Algorithm 1. Ensemble by vote

Require: T	▷ Text
Ensure: P_{final}	▷ Final prediction
1: procedure ENSEMBLE(T)	
2: $p_1 = Predict_{model-1}(T)$	
3: $p_2 = Predict_{model-2}(T)$	
4: $p_3 = Predict_{model-3}(T)$	
5: $p_{final} = mode(p_i)$	▷ $i = 1..3$
6: return P_{final}	
7: end procedure	

3.4. Evaluation metrics

In our studies, we employed two critical assessment criteria to evaluate the model’s performance:

- Accuracy: the ratio of accurately predicted labels to the total number of labels, serving as a comprehensive indicator of the model’s categorization efficacy.
- F1-score (weighted): the weighted F1-score considers both accuracy and recall, rendering it more appropriate for datasets with imbalanced classes. The weighted average F1-score is especially effective for assessing performance across all classes for our classifications challenge

3.5. Experimental setup, training and testing

For our experiment, we divided the dataset into training and testing sets using an 80-20 split, meaning 80% of the data is allocated for training the model. At the same time, the remaining 20% is reserved for testing its performance. The model was trained with 128 BiLSTM layers with a learning rate of 0.001. We used ‘Adam’ optimizer and ‘CrossEntropyLoss’ loss function. The model was trained for 300 epochs. During each epoch, the loss, training accuracy, and F1-score were tracked, and the model was evaluated on the test set to monitor its performance. We have used a standard scaler to normalize numerical data, referred to as AF. We measured loss values, accuracy, and F1-scores for every epoch along with Plots of loss values, testing, training, and F1-scores.

4. RESULTS AND ANALYSIS

This study explored the efficacy of integrating pre-trained language models (BERT and GPT) with POS tagging and other additional features in identifying rumors on social media. Previous studies have examined individual models such as BERT for text classification problems. However, they have not specifically considered GPT embedding and the benefits of incorporating supplementary features (e.g., ‘retweet.count’, ‘user.followers.count’, ‘user.friends.count’, and ‘favorite.count’) with ensemble learning to enhance the performance of the rumor detection model. In this section, we discuss the performance of the different models along with the proposed ensemble model (model-4) in terms of accuracy and F1-score on PHEME and translated Weibo datasets. The results indicated that the ensemble model consistently outperformed the individual models across all metrics. The hard voting mechanism effectively combined the strengths of the different models, leading to a more accurate and reliable rumor detection system. However, for Weibo, it is not properly justified to compare the performance with other systems, as we have used a translated dataset in English.

Table 2 shows a comparison between all the models along with our ensemble model, which proves that the ensemble model gave an average accuracy of 97.6% on PHEME datasets and 98.4% on WeiboE dataset. The bold values show the best performance for an individual model on these datasets. As we have seen, among the three models, four events of PHEME datasets give the best results for the POS+AF+GPT+BiLSTM model or model-3, whereas for one event (Ferguson), the GPT+AF+BiLSTM model or model-1 gives the best results. Model-3 gives the best result on the WeiboE dataset. The integration of additional characteristics and POS tags enhanced generalization in the task. The proposed ensemble model, or model-4, combining GPT+AF+BiLSTM (model-1), POS+BERT+AF+BiLSTM (model-2), and POS+GPT+AF+BiLSTM (model-3) excels in rumor detection by leveraging semantic and syntactic features, achieving superior accuracy and robustness compared to individual models.

Table 2. Comparison between baseline models with other variants. OT: Ottawashooting, GC: Germanwings crash, FE: Ferguson, CH: Charlihebd, SY: Sydneyside

Models	PHEME					WeiboE
	OT	GC	FC	CH	SY	
Baseline	Acc=84.8% F1=84.8%	Acc=86.2% F1=86.2%	Acc=86.2% F1=86.7%	Acc=88.5% F1=88.4%	Acc=85.7% F1=85.7%	Acc=92.8% F1=92.8%
Model-1	Acc=89.3% F1=89.3%	Acc=85.1% F1=85.1%	Acc= 86.9% F1= 86.9%	Acc= 92.5% F1= 92.4%	Acc=84.9% F1=84.9%	Acc=93.9% F1=93.8%
Model-2	Acc=83.7% F1=83.2%	Acc=81.9% F1=81.9%	Acc=82.9% F1=83.3%	Acc=89.2% F1=89.2%	Acc=84.9% F1=84.9%	Acc=93.2% F1=93.2%
Model-3	Acc= 89.3% F1= 89.3%	Acc= 87.2% F1= 87.2%	Acc=86.4% F1=86.6%	Acc= 92.6% F1= 92.4%	Acc= 86.2% F1= 86.1%	Acc= 94.3% F1=93.9%
Model-4 (proposed)	Acc= 97.5% F1= 97.4%	Acc= 97.8% F1= 97.7%	Acc= 97.3% F1= 97.2%	Acc= 98.4% F1= 98.3%	Acc= 97.1% F1= 97.1%	Acc= 98.4% F1= 98.3%

Tables 3 and 4 shows the comparison with the similar models in terms of accuracy and F1-score on PHEME and WeiboE datasets. Our ensemble model-4 outperformed similar systems by a great percentage. Integrating diverse neural architectures mitigates individual weaknesses and enhances generalization across varied datasets. However, this approach entails significant computational overhead and complexity, raising challenges in real-time applications and model maintenance. Over-fitting risks and dependency on high-quality training data are notable concerns, along with difficulties in interpretability and debugging. Despite these chal-

lenges, the model's high performance on benchmarks highlights its potential, necessitating further optimization and validation for practical deployment. Our findings corroborate prior research indicating the efficacy of GPT and BERT models in processing textual data; however, our ensemble model revealed that integrating additional metadata and applying hard voting can significantly improve classification performance.

Table 3. Comparison between different models on PHEME datasets

Model	Accuracy	F1-score
gDART [30]	94.8%	89.7%
RDLNP [31]	88.6%	88.6%
CNN-IG-ACO NB [32]	73.28%	73.2%
BiLSTM-CNN [33]	86.1%	86.1%
BERT+BiLSTM [25], [26]	86.2%	86.1%
Model-4 (proposed)	97.6%	97.5%

Table 4. Comparison between different models on Weibo datasets

Model	Accuracy	F1-score
VAE-GCN [34]	94.1%	94.0%
PostCom2DR [35]	95.0%	95.0%
Bi-GCN [36]	96.0%	96.0%
DDGCN [37]	94.8%	95.2%
Model-4 (proposed)	98.4%	98.3%

Note: We have used WeiboE (translated in English)

In this work, we assessed the efficacy of three distinct models model-1, model-2, and model-3 each employing varied feature sets and topologies. Model-1 amalgamated GPT embeddings, AF, and BiLSTM; model-2 employed POS tagging, BERT embeddings, AF, and BiLSTM; model-3 incorporated POS tagging, GPT embeddings, AF, and BiLSTM. Their performance fluctuated, with accuracy between 84.8% and 92.6% and F1-scores from 84.8% to 92.4% for PHEME dataset of different datasets and accuracy from to 92.8% to 94.3% with F1-score from 92.8% to 93.9% for Weibo dataset. The proposed ensemble model, which consolidates predictions from these models through hard voting, attained an overall accuracy of 97.6% and an F1-score of 97.5% for the PHEME dataset and an accuracy of 98.4% and an F1-score of 98.3% for the WeiboE dataset, illustrating substantial enhancement by utilizing the strengths of each model and improving overall classification robustness. The proposed ensemble model integrating GPT, BERT, POS tagging, and supplementary characteristics exhibited enhanced performance compared to individual models. The results strongly indicate that integrating diverse information types enhances rumor identification, with potential applications in real-time monitoring systems. Although the ensemble model showed strong performance, this work concentrated mostly on a dataset of tweets, perhaps constraining the applicability of the findings to other types of textual data. The computational cost associated with training extensive models such as GPT and BERT may provide a constraint for real-time applications.

5. CONCLUSION

Our work advances rumor detection research by integrating POS tagging, additional features, and BERT or GPT-based embeddings with BiLSTM networks using the standard PHEME and Wiebo datasets. We developed three predictive models by combining these features and ultimately proposed an ensemble method. This approach aims to leverage the strengths of individual models into an ensemble, resulting in a robust and accurate rumor detection system. Our method addresses the limitations of traditional techniques and standalone deep learning models, offering a comprehensive solution. Through various experimental studies, indeed, it is clear that our ensemble method ranks better than other methods in terms of the accuracy of rumor detection. Through the use of multiple models that encode different aspects of the language and context, the possibility of certain methods' deficiencies reflecting on the final output is significantly reduced. More work is planned to be done in the future including the examination of other types of ensemble algorithms, e.g. soft voting or stacking, as well as including new features such as temporal data or metadata about the users in order to improve detection rates.




REFERENCES

- [1] A. Zubiaga, M. Liakata, R. Procter, G. W. S. Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PLoS ONE*, vol. 11, no. 3, p. e0150989, Nov. 2016, doi: 10.1371/journal.pone.0150989.
- [2] J. Ma *et al.*, "Detecting rumors from microblogs with recurrent neural networks," in *IJCAI International Joint Conference on Artificial Intelligence*, 2016, pp. 3818–3824.
- [3] M. Celliers and M. Hattingh, "A systematic review on fake news themes reported in literature," in *Responsible Design, Implementation and Use of Information and Communication Technology: 19th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2020*, 2020, vol. 12067 LNCS, pp. 223–234, doi: 10.1007/978-3-030-45002-1_19.
- [4] X. Zhou and R. Zafarani, "A survey of fake news: fundamental theories, detection methods, and opportunities," *ACM Computing Surveys*, vol. 53, no. 5, pp. 1–40, Sep. 2021, doi: 10.1145/3395046.
- [5] D. de Beer and M. Matthee, "Approaches to identify fake news: a systematic literature review," *Integrated science in digital age 2020*, vol. 136, pp. 13–22, 2021, doi: 10.1007/978-3-030-49264-9_2.
- [6] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and resolution of rumours in social media," *ACM Computing Surveys*, vol. 51, no. 2, pp. 1–36, Mar. 2018, doi: 10.1145/3161603.
- [7] M. Al-Sarem, W. Boulila, M. Al-Harby, J. Qadir, and A. Alsaedi, "Deep learning-based rumor detection on microblogging platforms: a systematic review," *IEEE Access*, vol. 7, pp. 152788–152812, 2019, doi: 10.1109/ACCESS.2019.2947855.
- [8] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," *Information Sciences*, vol. 497, pp. 38–55, Sep. 2019, doi: 10.1016/j.ins.2019.05.035.
- [9] D. Varshney and D. K. Vishwakarma, "A review on rumour prediction and veracity assessment in online social network," *Expert Systems with Applications*, vol. 168, p. 114208, Apr. 2021, doi: 10.1016/j.eswa.2020.114208.
- [10] L. Tan, G. Wang, F. Jia, and X. Lian, "Research status of deep learning methods for rumor detection," *Multimedia Tools and Applications*, vol. 82, no. 2, pp. 2941–2982, Jan. 2023, doi: 10.1007/s11042-022-12800-8.
- [11] B. Pattanaik, S. Mandal, and R. M. Tripathy, "A survey on rumor detection and prevention in social media using deep learning," *Knowledge and Information Systems*, vol. 65, no. 10, pp. 3839–3880, Oct. 2023, doi: 10.1007/s10115-023-01902-w.
- [12] C. Castillo, M. Mendoza, and B. Poblete, "Information credibility on twitter," in *Proceedings of the 20th international conference on World Wide Web*, Mar. 2011, pp. 675–684, doi: 10.1145/1963405.1963500.
- [13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [14] N. Ruchansky, S. Seo, and Y. Liu, "CSI: a hybrid deep model for fake news," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 797–806.
- [15] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, "Object recognition with gradient-based learning," in *Shape, Contour and Grouping in Computer Vision*, Springer, 1999.
- [16] J. Devlin, M.-W. Chang, K. Lee, K. T. Google, and A. I. Language, "BERT: pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [17] R. Anggrainingsih, G. M. Hassan, and A. Datta, "BERT based classification system for detecting rumours on Twitter," *arXiv preprint arXiv:2109.02975*, 2021, [Online]. Available: <http://arxiv.org/abs/2109.02975>.
- [18] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2018.
- [19] Q. Liu, X. Tao, J. Wu, S. Wu, and L. Wang, "Can large language models detect rumors on social media?," *arXiv preprint arXiv:2402.03916*, 2024, [Online]. Available: <http://arxiv.org/abs/2402.03916>.
- [20] A. Chakraborty, S. Joardar, and A. A. Sekh, "Ensemble classifier for Hindi hostile content detection," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 23, no. 1, pp. 1–17, 2024, doi: 10.1145/3591353.
- [21] C. M. M. Kotteti, X. Dong, and L. Qian, "Ensemble deep learning on time-series representation of tweets for rumor detection in social media," *Applied Sciences (Switzerland)*, vol. 10, no. 21, pp. 1–21, 2020, doi: 10.3390/app10217541.
- [22] L. Yuan, J. Wang, and X. Zhang, "YNU-HPCC at SemEval-2020 Task 8: using a parallel-channel model for memotion analysis," in *14th International Workshops on Semantic Evaluation, SemEval 2020 - co-located 28th International Conference on Computational Linguistics, COLING 2020, Proceedings*, 2020, pp. 916–921, doi: 10.18653/v1/2020.semeval-1.116.
- [23] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2014, pp. 1724–1734, doi: 10.3115/v1/d14-1179.
- [24] K. Nithya, M. Krishnamoorthi, S. V. Easwaramoorthy, C. R. Dhivyaa, S. Yoo, and J. Cho, "Hybrid approach of deep feature extraction using BERT–OPCNN & FIAC with customized Bi-LSTM for rumor text classification," *Alexandria Engineering Journal*, vol. 90, pp. 65–75, 2024, doi: 10.1016/j.aej.2024.01.056.
- [25] R. Cai *et al.*, "Sentiment analysis about investors and consumers in energy market based on BERT-BiLSTM," *IEEE Access*, vol. 8, pp. 171408–171415, 2020, doi: 10.1109/ACCESS.2020.3024750.
- [26] Z. Zhu and L. Wang, "BERT-BiLSTM model for entity recognition in clinical text," *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2022) co-located with the Conference of the Spanish Society for Natural Language Processing (SEPLN 2022)*, vol. 3202, 2022.
- [27] B. Pattanaik, "WeiboE," *GitHub*, 2024. <https://github.com/barshapattanaik/WeiboE> (accessed Jun. 27, 2024).
- [28] "Gpt-2 documentation," *Hugging Face*. https://huggingface.co/docs/transformers/main/en/model_doc/gpt2 (accessed Nov. 29, 2024).
- [29] M. M. Rahman, Y. Watanobe, and K. Nakamura, "A bidirectional LSTM language model for code evaluation and repair," *Symmetry*, vol. 13, no. 2, p. 247, 2021, doi: 10.3390/sym13020247.
- [30] S. Roy, M. Bhanu, S. Saxena, S. Dandapat, and J. Chandra, "gDART: improving rumor verification in social media with discrete attention representations," *Information Processing & Management*, vol. 59, no. 3, p. 102927, May 2022, doi: 10.1016/j.ipm.2022.102927.
- [31] A. Lao, C. Shi, and Y. Yang, "Rumor detection with field of linear and non-linear propagation," in *Proceedings of the Web Conference 2021*, Apr. 2021, pp. 3178–3187, doi: 10.1145/3442381.3450016.




- [32] A. Kumar, M. P. S. Bhatia, and S. R. Sangwan, "Rumour detection using deep learning and filter-wrapper feature selection in benchmark twitter dataset," *Multimedia Tools and Applications*, vol. 81, no. 24, pp. 34615–34632, Oct. 2022, doi: 10.1007/s11042-021-11340-x.
- [33] M. Z. Asghar, A. Habib, A. Habib, A. Khan, R. Ali, and A. Khattak, "Exploring deep neural networks for rumor detection," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 4, pp. 4315–4333, Apr. 2021, doi: 10.1007/s12652-019-01527-4.
- [34] H. Lin, X. Zhang, and X. Fu, "A graph convolutional encoder and decoder model for rumor detection," in *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, Oct. 2020, pp. 300–306, doi: 10.1109/DSAA49011.2020.00043.
- [35] Y. Yang, Y. Wang, L. Wang, and J. Meng, "PostCom2DR: utilizing information from post and comments to detect rumors," *Expert Systems with Applications*, vol. 189, p. 116071, Mar. 2022, doi: 10.1016/j.eswa.2021.116071.
- [36] T. Bian *et al.*, "Rumor detection on social media with bi-directional graph convolutional networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 549–556, Apr. 2020, doi: 10.1609/aaai.v34i01.5393.
- [37] M. Sun, X. Zhang, J. Zheng, and G. Ma, "DDGCN: dual dynamic graph convolutional networks for rumor detection on social media," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, pp. 4611–4619, Jun. 2022, doi: 10.1609/aaai.v36i4.20385.

BIOGRAPHIES OF AUTHORS






Barsha Pattanaik    is a Ph.D. scholar at the School of Computer Science and Engineering (SCSE) at XIM University in Bhubaneswar, Odisha, India. Her work focuses on DL and NLP, especially in building different language models and using them to detect rumors on social media. She holds a Master's degree in Electronics and telecommunication engineering from BPUT University, Odisha. Her current research aims to build more robust models for the detection of rumor content in social media contexts. She can be contacted at email: barsha@xustudent.edu.in.






Sourav Mandal    has been an assistant professor at the School of Computer Science and Engineering (SCSE) at XIM University in Bhubaneswar, Odisha, India, since October 2020. Previously, he served as an assistant professor in the Department of Computer Science and Engineering at Haldia Institute of Technology in Haldia, India, beginning in 2006. His research interests span NLP, computer vision, and image processing, with a focus on information extraction, text and image classification, sentiment analysis, large language models (LLM) and generative texts, multimodal data, and meme analysis. He earned a bachelor's degree in Computer Science and Engineering from The University of Burdwan in 2003, a master's degree in multimedia development from Jadavpur University in 2005, and a Ph.D. in engineering from Jadavpur University in 2020. He has published numerous research articles in indexed journals and conference proceedings. He can be contacted at email: sourav@xim.edu.in.



Rudra Mohan Tripathy    is currently working as an associate professor and Dean (Academic) in the School of Computer Science and Engineering, XIM University, Bhubaneswar. He has more than 22 years of teaching experience in various reputed institutes. He served as Head of the Department of Computer Science and Engineering, Silicon Institute of Technology for more than 6 years. He holds a Ph.D. in Computer Science and Engineering from IIT Delhi. His research focuses on the areas of data mining, machine learning, social network analysis, and structural properties of networks. His research work on "Rumor Control Strategies" has been covered by many news media: The Hindu, The Indian Express, BBS Fake News. He has published many papers in reputed international conferences and journals. He can be contacted at email: rudramohan@xim.edu.in.



Arif Ahmed Sekh    is a senior researcher at UiT The Arctic University of Norway. Previously, he was an assistant professor at the School of Computer Science and Engineering, XIM University, Bhubaneswar, India (2021-2024). Before that, he was post-doctoral research fellow in the Department of Physics and Technology at the UiT The Arctic University of Norway, Tromsø, Norway (2019-2021). From 2009 to 2019, he was an Assistant Professor of Computer Application at Haldia Institute of Technology, India. He is a senior member of IEEE and Digital Life Norway (DLN). He can be contacted at email: skarifahmed@gmail.com.