

Unsupervised Word Embedding with Ensemble Deep Learning for Twitter Rumor Identification

Devaraj Rajasekar*, Lourdusamy Robert

Department of Computer Science, Government Arts College, Coimbatore -641018, India

Corresponding Author Email: rsekar.phd@gmail.com



<https://doi.org/10.18280/ria.360515>

ABSTRACT

Received: 27 July 2022

Accepted: 18 October 2022

Keywords:

rumor identification, EGCN, word-embedding, unsupervised learning, n-gram attributes

In social networks, rumor identification is a major problem. The structural data in a topic is applied to derive useful attributes for rumor identification. Most standard rumor identification methods concentrate on local structural attributes, ignoring the global structural attributes that exist between the source tweet and its responses. To tackle this issue, a Source-Replies relation Graph (SR-graph) has been built to develop an Ensemble Graph Convolutional neural Net (EGCN) with a Nodes Proportion Allocation Mechanism (NPAM) which identifies the rumor. But, the word vectors were trained by the standard word-embedding model which does not increase the accuracy for large Twitter databases. To solve this problem, an unsupervised word-embedding method is needed for large Twitter corpora. As a result, the Twitter word-embedded EGCN (T-EGCN) model is proposed in this article, which uses unsupervised learning-based word embedding to find rumors in huge Twitter databases. Initially, the latent contextual semantic correlation and co-occurrence statistical attributes among words in tweets are extracted. Then, to create a rumor attribute vector of tweets, these word embeddings are concatenated with the GloVe model's word attribute vectors, Twitter-specific attributes, and n-gram attributes. Further, the EGCN is trained by using this attribute vector to identify rumors in a huge Twitter database. Finally, the testing results exhibit that the T-EGCN achieves 87.56% accuracy, whereas the RNN, GCN, PGNN, EGCN, and BiLSTM-CNN attain 65.38%, 68.41%, 75.04%, 81.87%, and 86.12%, respectively for rumor identification.

1. INTRODUCTION

Microblogging websites are valuable data sources that have been effectively used to analyze sociopragmatic phenomena including faith, comment, and emotions in a digital environment. Microblogs are the ones to report breaking news before it reaches corporate media channels [1]. Twitter is one of the most important platforms for investigating Natural Language Processing (NLP). The presence of such enormous amounts of information is a double-edged sword. It is easy to obtain untrustworthy data from certain sources, and it is difficult to regulate the dissemination of inaccurate data [2].

It is difficult to identify between true and rumor (inaccurate data), particularly when the information seems to be prepared and properly organized. This is extremely important in emergencies [3]. For instance, by just clicking the Re-tweet button on Twitter, a bit of information may quickly become popular. The phrase "rumor" has multiple interpretations. A rumor might be real or untrue. It is a claim of questionable veracity and no obvious source, even if its ideological or political roots and intentions are evident [4].

Rumor is a significant phenomenon in sociology that has fascinated the attention of several experts in the field of sociology over certain eras. It seems to be a more important systemic issue because of the fast rise in broad social networking sites like Facebook, Twitter, and Sina Weibo [5, 6]. Due to the ease of obtaining data on various digital platforms, they can disseminate quickly before being corrected

or noticed. Rumor is a bit of data whose veracity is questioned, whose source is questionable, and which is likely to appear in an unexpected crisis, spreading fear among the population, undermining the government's confidence, disrupting society, and even endangering global defense. Rumors can cause feelings like stress and worry, which impairs people's ability to make sensible decisions [7]. For instance, on March 12, 2011, after a strong earthquake in Japan, a rumor had been widespread across China through Sina Weibo and other social networks that iodized salt might protect individuals from nuclear radiation. Therefore, individuals rushed to markets, shops, and dispensaries to buy salt. During that period, the iodized salt cost had increased by 5-10%. So, it is a crucial process to identify and report rumors from social networks before they spread.

The fundamental distinction between rumors and fake news is that rumor is unproven data that may come out to be real or untrue or may stay unaddressed [8], whereas fake news is incorrect data that is frequently propagated via news channels. Because rumor and fake news have so many similarities in terms of features, rumor identification is quite comparable to fake news identification methods. Rumor identification frequently aims to discriminate between proven and unproven data [9]. The goal of fake news identification is to differentiate between real and deceptive news. The spread of incorrect information on the internet has the potential to agitate people's behavior, influence popular sentiment, and, more importantly, destroy the administration's legitimacy and instigate social

unrest [10, 11].

Nonetheless, rumor identification is a difficult process because of the three main factors: i). the difficult task of processing large amounts of data. On social media, breaking news about current events appears incessantly, and the material spans a wide range of topics.

To detect various types of rumors, a wide range of data must be evaluated. ii). the requirement for real-time identification. Individuals on online platforms are energetic, providing a broad range of data to circulate quickly. Because the negative effects of false rumors grow rapidly, it is critical to recognize rumors as soon as necessary [12]; and iii). the perplexing behavior of rumors. Certain rumors are purposefully made to look like actual news for a variety of reasons, including social astroturfing and malevolent sales operations [13]. Individuals, including field specialists, have difficulty distinguishing between real and misleading rumors.

Various research on rumor identification models has been conducted to detect rumors on social media [14-17]. These models employ different machine learning algorithms such as Random Forest (RF), Support Vector Machine (SVM), etc., to mine the content attributes and context attributes for rumor identification [18]. The content attributes comprise syntactic, lexical, and semantic attributes whereas the context attributes represent structural attributes. Nevertheless, several classical models concentrate on local structural attributes whilst the global structural attributes between the source tweet and its replies were not well utilized. Also, the content and local structural attributes make learning complicated.

To combat these challenges, an unsupervised learning model is essential to extract more attributes and identify rumors from the large-scale Twitter corpus. Therefore in this manuscript, the T-EGCN model is proposed which applies an unsupervised learning-based word embedding to identify rumors from large Twitter databases. In this model, the latent contextual semantic correlation and co-occurrence statistical features among words in tweets are utilized. Then, these word embeddings are concatenated with the word attribute vectors from the GloVe model, Twitter-specific attributes, and n-gram attributes to generate a rumor attribute vector of tweets. Moreover, this attribute vector is learned by the EGCN to identify rumors from a large Twitter database. Thus, this can increase the detection accuracy by learning more attributes between the source tweet and its replies. The remaining sections are arranged as follows: Section II reviews the recent rumor identification models suggested by different authors. Section III explains the T-EGCN model and Section IV illustrates its performance. Section V summarizes the entire work and gives future directions.

2. LITERATURE SURVEY

Bai et al. [19] designed the SR-graph which utilizes the global structural attributes and content data entirely. By using the SR-graphs, an EGCN with NPAM was developed to identify the rumor. This model was used to learn text, local, and global structural attributes for rumor identification. Also, the word vector dimensions were optimized to get acceptable efficiency. On the other hand, the word vectors were learned by using the classical word-embedding model which results in poor performance for large Twitter databases. So, it needs to design an unsupervised word-embedding method for large Twitter corpora.

Alzanin and Azmi [20] investigated the challenge of identifying rumors in Arabic tweets. First, the collection of tweets was obtained and pre-processed to eliminate inappropriate tweets. Then, the attributes were mined from the client and the content to measure their importance. Further, a model called semi-supervised Expectation-Maximization (EM) was applied to learn such attributes to identify rumors. But the accuracy was degraded due to the unlabeled data.

Fard et al. [21] developed a novel One-Class Categorization (OCC) to recognize the computational rumors in social media. In this OCC, two principles were considered to mine the attributes: the primary one concentrated on prior accessible attributes and the second one considered attributes that have been recommended as useful attributes for identifying rumors in social networks. On the other hand, it needs more attributes influenced by the background of the rumor to increase efficiency.

Wu et al. [22] developed a Propagation Graph Neural Network (PGNN) which creates effective interpretations for all nodes in the PG to detect the rumor. In the PGNN structure, the node interpretations were continuously modified by transferring data among the adjacent nodes through relation routes within limited intervals. Different varieties of PGNN such as Global embedding with PGNN (GLO-PGNN) and Ensemble learning with PGNN (ENS-PGNN) were developed. Also, an attention strategy was applied to adaptively fine-tune the weight of all nodes. But, it has several learning variables and so the variable fine-tuning was tedious.

Alkhodair et al. [23] presented a novel system that together trains word embeddings and Recurrent Neural Networks (RNN) to identify rumors. In this system, a novel semi-supervised training was applied by integrating unsupervised and supervised training objectives to detect the breaking news rumor. Also, a novel policy was used to modify word embeddings with the learning task for preventing cross-topic and out-of-vocabulary problems in identifying the breaking news rumor. But this system does not memorize the particulars across the interval.

Kotteti et al. [24] designed an ensemble framework that executes the majority-voting method on a set of estimations of NN by moment-sequence vector sign of Twitter information to identify the rumors. At first, the raw twitter talks were practiced to convert them into the desired format. Afterward, such information was provided to the ensemble framework to predict the rumor and non-rumor data. But it considers the attributes only dependent on the temporal property of the tweet, whereas it needs content and context-based attributes to further increase the efficiency.

Yu et al. [25] developed a new rumor recognition framework depending on GCN which signifies the dissemination pattern of rumors with a graph convolution operator for node vector fine-tuning. In this framework, static and dynamic attributes were mined from the original database from social media. Also, the attribute merging and pooling units were optimized to enhance efficiency. However, its accuracy was still not effective while considering the limited quantity of learning data.

Song et al. [26] designed a new rumor identification method called Credible Early Detection with Convolutional Neural Network (CED-CNN) to find an early point in time to make a reliable forecast about each repost to a rumored candidate as a sequence. But it cannot obtain a reliable forecast point and create a reliable forecast because the reposts were relatively ambiguous. Chen et al. [27] designed a new participant-level

rumor identification method that combines different fine-grained client interpretations of each participant from the propagation threads through deep representation training. However, the database considered was limited and a small fraction of client profiles were no longer accessible.

Asghar et al. [28] studied the rumor identification issue by exploring various deep learners with stress on accounting for the background data in forwarding and rearward orders for the passage. First, Bidirectional LSTM (BiLSTM) was applied to train the enduring dependence in a tweet by considering the precedent and potential background data. Then, CNN was employed for mining attributes to categorize the tweet into rumors and non-rumors. However, because it only considers text-based attributes, its efficiency is lower.

Choudhary and Arora [29] developed a linguistic feature-based learning framework to identify and categorize fake news. Initially, linguistic attributes were mined using different computational methods along with the syntactic, grammatical, sentimental, and readability attributes. Then, these attributes were learned by the Sequential Neural Network (SNN) to identify the fake and real news. But its accuracy was not effective and needs more attributes to enhance the system's efficiency.

2.1 Research contribution

From the literature, it is addressed that the accuracy of rumor identification models based on supervised learning is influenced by different factors such as unlabeled data, lack of content and context-based attributes, and more training parameters. This results in poor efficiency on large-scale Twitter databases. So, this research concentrates on improving the efficiency of Twitter rumor identification based on unsupervised learning using large-scale Twitter databases. The scientific contribution of this research is the following:

- First, a huge amount of tweets and their responses are collected.
- Then, different attributes are extracted and represented from the collected database using various schemes.
- Moreover, the extracted attributes of a tweet also known as its corresponding word vectors are merged as a unified word vector.
- Further, the unified word vector is learned by the EGCN for the rumor identification process.

3. PROPOSED METHODOLOGY

In this section, an overview of GCN and EGCN models is initially discussed separately. Then, an improvement of those models called the T-EGCN model is described.

3.1 Overview of GCN model

The GCN or Text GCN model has been developed as a text classification technique by Yao et al. [30]. In this study, a heterogeneous word document graph was constructed for an entire database and converted the document classification problem into the node classification problem. To solve this problem, the constructed graph was fed into the basic 2-layer GCN, which captures global word co-occurrence attributes (i.e., content attributes) and uses limited labeled documents for document/text classification. But, the main drawback of this model was that it was inherently transductive, in which test

document nodes (words without labels) were added to learning. So, this model was not able to rapidly create embeddings and provide a prediction for unlabeled test documents. Also, the content attributes were not adequate for text classification or rumor detection processes.

3.2 Overview of EGCN model

To enhance the text/rumor recognition performance the Text CNN model was incorporated with the GCN called EGCN using the NPAM, which utilizes global structural attributes and content attributes between the source tweets and their responses [19]. The rumor recognition problem was converted into the text classification problem. To solve this problem, the word vectors were learned by the Word2Vec model, and an SR-graph was created for each source tweet and its response. Then, the word vectors were fed to the Text CNN and the SR-graph was fed to the GCN for capturing high-level characteristics. Moreover, the output of those network models was combined by the NPAM and passed to the softmax layer to detect rumors. Nevertheless, it was considered the standard word-embedding model to learn the word vectors, resulting in poor efficiency for large Twitter databases.

To tackle this issue, this study proposes a T-EGCN model for rumor detection, which is discussed in the below section.

3.3 Proposed T-EGCN model for rumor detection

The proposed T-EGCN involves data pre-processing, tweet rumor attribute vector generation, SR-relation graph creation, and rumor detection.

3.3.1 Data pre-processing

The pre-processing of tweets involves the following steps:

- Eliminating every non-ASCII and non-English symbol in the tweets.
- Eliminating every URL link, integer, and stop word to refine the tweets.
- Substituting negative references: Tweets have distinct notions of negation. The task of negation is converting won't, can't, and n't into will not, cannot, and not, independently.
- Expanding acronyms and slang to their complete word format.
- Substituting emoticons and emoji's using their source text format by looking up the emoticon dictionary.

3.3.2 Tweet rumor attribute vector generation

In this study, different categories of attributes from each topic are extracted and unified to create a tweet rumor attribute vector.

Word n-gram attributes: These are highly useful for identifying rumors in social networking platforms. In this study, unigram and bigram attributes called Bag-of-Words (BoW) attributes are considered as the baseline attribute representations.

Twitter-specific attributes: The number of hashtags, emoticons, negation, Part-Of-Speech (POS), and the existence of capitalized words are known as Twitter-specific attributes.

Word representation attributes: The word vector representations from unsupervised training in the large database can obtain grammatical and semantic attributes of words. In this model, the GloVe framework is applied to

perform unsupervised training of word-level embeddings. It is a global log bilinear regression technique that integrates the benefits of local context window and global matrix factorization techniques. It uses statistical data by simply learning the non-zero factors in a word-word co-occurrence matrix instead of learning the complete sparse matrix or separate context windows in a huge database. It trains word vectors with percentages of co-occurrence chance instead of the chance itself.

Assume words w_a and w_b that reveal a certain part of attention. The correlation of such words is determined by analyzing the percentage of their co-occurrence chances with different probe words w_p . Consider \mathcal{P}_{ab} is the chance that word b exist in the context of word w_a . Then, they are mapped to attribute vectors that are near to every other since synonyms and related paragraphs normally include analogous context. Once the GloVe framework is trained, the word vectors are defined as the semantic attributes of the tweet. Moreover, these vectors are merged with the weighted word vectors as a unified attribute vector called the tweet rumor attribute vector. So, the GloVe algorithm is applied to learn tweet word vectors in a large Twitter database.

3.3.3 SR graph for extracting structural characteristics

The SR-graph for each tweet is constructed to get the global structural data and retrieve the global and local attributes. Also, the rumor detection process is converted into the graph classification process. In the SR graph, the neighboring matrix can reveal the global structure of a topic. When a tweet responds to the other tweet, its related nodes are linked. The node attributes in an SR-graph are defined by the weighted word vectors wherein the node weight is the ratio of its extent. The weights are utilized to reveal the local structure of a topic and the Word2Vec algorithm is used to learn word vectors. Also, unique characters, websites, and other texts are converted to word vectors. As a part of input data for neural nets, all word vectors are normalized to $[0, 1]$. Figure 1 depicts an example of the SR relation graph created for tweets with 6 responses.

3.3.4 NPAM

The NPAM is applied to create an ensemble network for various topics. The ensemble network comprises a Text CNN and the GCN. Considering that the number of nodes in the current SR-graph is N , the number of nodes in the maximal SR-graph is M , the contribution rate of the Text-CNN and the GCN for classification is described by $\frac{N}{M}$. The resultant deep ensemble network is called the T-EGCN.

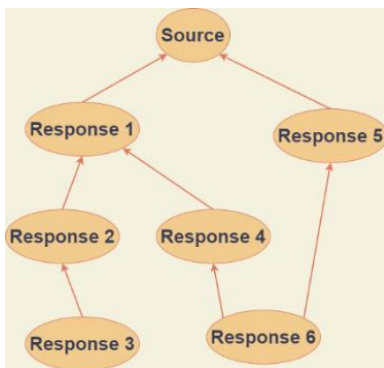


Figure 1. Diagrammatic representation of SR relation graph with 6 responses

3.3.5 Rumor detection

The features extracted from SR-graphs and the tweet rumor attribute vector is converted into the binary classification process for rumor detection. In this study, the tweet rumor attribute vectors and the correlation between the source and its responses are crucial for rumor detection. For all tweets and their responses, their corresponding tweet rumor attribute vector and SR-graph are obtained. The tweet rumor attribute vector is fed to the Text CNN and the SR graph is fed to the GCN for learning high-level characteristics. Then, the NPAM is applied to create an ensemble network for various topics. Figure 2 illustrates the architecture of the T-EGCN model for the rumor identification process.

As illustrated in Figure 2, considering that the input of the T-EGCN is a topic, the corresponding tweet rumor attribute vector and the SR-graph of this topic are obtained. The SR graphs and the corresponding tweet rumor attribute vectors are fed to the GCN and Text-CNN models, respectively.

The Text-CNN has 3 units, whereas the GCN comprises 4 graph convolution units, a SortPooling unit, and 3 1D convolution units. The result of the T-EGCN is an ensemble of the Text-CNN and the GCN models. The convolution process of the Text-CNN is defined by:

$$Y = \sum_{i=1} (W^{(i)} \times x) + b^{(i)} \quad (1)$$

In Eq. (1), $W^{(i)}$ is i^{th} convolution kernel, which is optimized by the back-propagation algorithm. Once the convolution process is completed, the Rectified Linear Unit (ReLU) activation is utilized.

$$Y_{conv} = ReLU(Y) = max(0, Y) \quad (2)$$

After convolution units, the attributes like keywords were obtained and the higher-level attributes are obtained by the pooling units. The aggregated attribute is defined by:

$$Y_{pool} = Pool_{max}(Y_{conv}) \quad (3)$$

In Eq. (3), $Pool_{max}$ is the max-pooling process, the attributes retrieved by the pooling units are fed to the fully-connected unit.

In the GCN, for a graph G and its node attributes A , the graph convolution unit creates the form:

$$X = f(\tilde{D}^{-1} \tilde{G} A W) \quad (4)$$

In Eq. (4), $\tilde{G} = G + 1$ denotes the neighboring matrix of the graph with self-loops, \tilde{D} denotes its diagonal degree matrix with $\tilde{D}_{ij} = \sum_j \tilde{G}_{ij}$, W denotes the matrix of learnable graph variables, f indicates the nonlinear activation function, and X denotes the resultant activation matrix. Then, multiple graph convolution units are stacked as:

$$X^{t+1} = f(\tilde{D}^{-1} \tilde{G} X^t W^t) \quad (5)$$

After many graph convolution units, a SortPooling unit is utilized to rank the attribute descriptors, every of which defines a vertex. The SortPooling process describes a series of nodes in the graph. After that, the outcome of this unit is fed to the fully connected unit.

The attribute outcome of the Text-CNN is proportional to the rate of $\frac{N}{M}$. Considering that the attribute outcome of the Text-CNN is α and the attribute outcome of the GCN is β , the complete attribute outcome of the T-EGCN is an ensemble of α and β using the NPAM as follows:

$$y = \alpha \times \frac{N}{M} + \beta \left(1 - \frac{N}{M}\right) \quad (6)$$

Moreover, the complete attribute vector y is given to the softmax unit, which classifies them into either the given input topic is rumor or non-rumor.

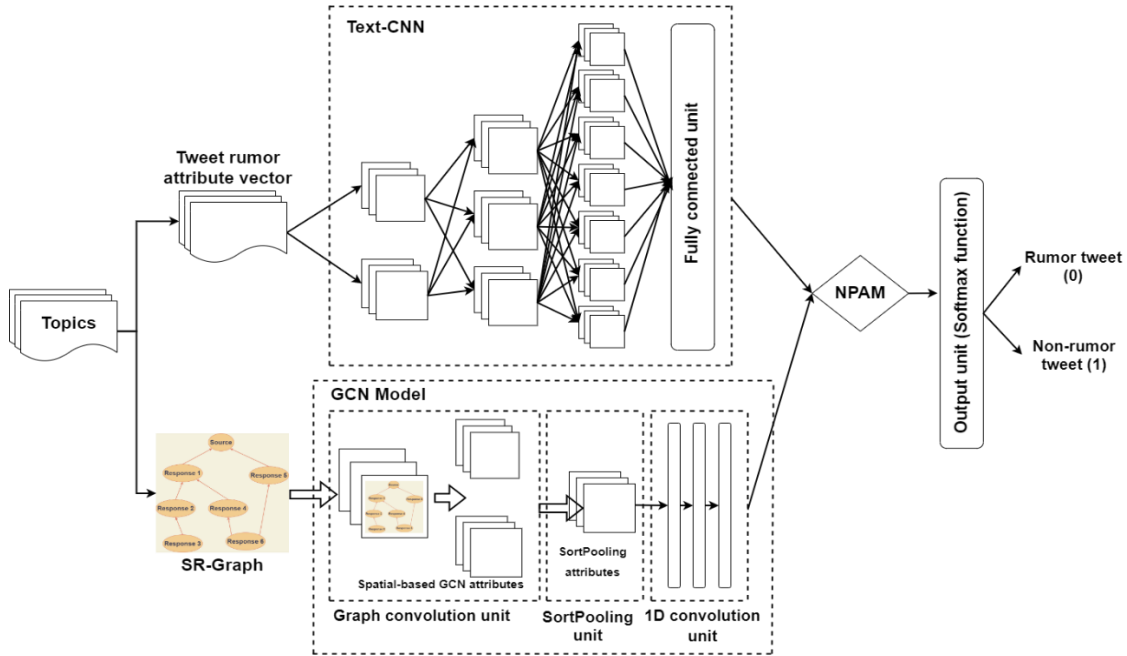


Figure 2. Architecture of T-EGCN

Algorithm for T-EGCN Model for Rumor Identification:

Input: PHEME database

Output: Rumor and non-rumor tweets

Begin

Collect and preprocess the number of tweets and their responses;

Split the database into training and test sets;

for(training set)

Construct the SR-graph;

Define the local and global structural attributes;

Obtain a variety of attributes such as n-gram attributes, Twitter-specific attributes, grammatical, and semantic attributes of words;

Train the GloVe and Word2Vec algorithms to get the final word vectors;

Combine different word vectors into a unified one called tweet rumor attribute vector;

Implement T-EGCN with the NPAM to learn the SR-graph and corresponding tweet rumor attribute vectors for various topics;

end for

Validate the trained T-EGCN with the NPAM using a test set;

Identify rumor and non-rumor tweets;

End

4. EXPERIMENTAL RESULTS

In this section, the efficiency of the T-EGCN model is analyzed by implementing it in Java. Also, its efficiency is compared with the classical models: EGCN [19], PGNN [22], RNN [23], GCN [25] and BiLSTM-CNN [28]. The

comparative analysis is conducted regarding accuracy, precision, recall, and f-measure.

4.1 Hyperparameter initialization

In this experiment, considering that the maximum parameter of the Text-CNN is the tweet rumor attribute vector, it is essential to feed the network with high-quality embeddings. So, Word2Vec neural framework is utilized to learn the word embeddings on unsupervised tweets. To learn the word embeddings, a skip-gram model with a window size of 5 is used and words with a frequency less than 5 are filtered. The weighted 25-D word vectors are used as the node attributes in SR graphs.

For Text-CNN training, the number of convolution units is 6, the number of weights of the convolution units is 32,425, the number of biases of the convolution units is 94, the size of kernels used in the convolution units is 11, the number of kernels is 94, the learning rate is 0.001, the number of neurons in the fully connected unit is 1000, the batch size is 54, the number of an epoch is 250 and the optimizer is Adam.

For the GCN training, 2-layer GCN is considered. The dropout rate for each layer is 0.5, the number of hidden units is 64, the maximum number of an epoch is 250, the optimizer is Adam, the learning rate is 0.001 and the L_2 -normalization factor is 10^{-5} .

4.2 Dataset description

In this analysis, the extended PHEME database is utilized which holds rumor and non-rumor Twitter posts during breaking news [31]. It comprises rumors associated with 9 events and all the rumors are annotated with their veracity

range: truthful, fake, or unconfirmed. Table 1 presents the number of all events in this database and the label distribution for the rumor identification process.

This T-EGCN model is experimented with using the top 5 major events which generate a highly balanced database. The 5 major events are Charlie Hebdo, Ferguson, Gencrash, Ottawa shooting, and Sydney Siege. So, the resultant database comprises 5802 annotated tweets, of which 1972 (34%) are identified as rumors and 3830 (66%) are non-rumors.

Table 1. Statistics of PHEME database

Events	#Threads	#Tweets	#Rumors	#Non-rumors
Charlie Hebdo	2079	38268	458 (22%)	1621 (78%)
Sydney siege	1221	23996	522 (42.8%)	699 (57.2%)
Ferguson	1143	24175	284 (24.8%)	859 (75.2%)
Ottawa shooting	890	12284	470 (52.8%)	420 (47.2%)
Germanwings-crash	469	4489	238 (50.7%)	231 (49.3%)
Putin missing	238	832	126 (52.9%)	112 (47.1%)
Prince Toronto	233	902	229 (98.3%)	4 (1.7%)
Gurllitt	138	179	61 (44.2%)	77 (55.8%)
Ebola Essien	14	226	14 (100%)	0 (0%)
Overall	6425	105354	2402 (37.4%)	4023 (62.6%)

4.3 Accuracy

It determines the fraction of accurate identifications over the total number of tweets analyzed.

$$Accuracy = \frac{True\ Positive\ (TP) + True\ Negative\ (TN)}{TP + TN + False\ Positive\ (FP) + False\ Negative\ (FN)} \quad (7)$$

Here, TP and FN are the number of tweets that are accurately identified. Also, FP and TN are the number of tweets that are inaccurately identified.

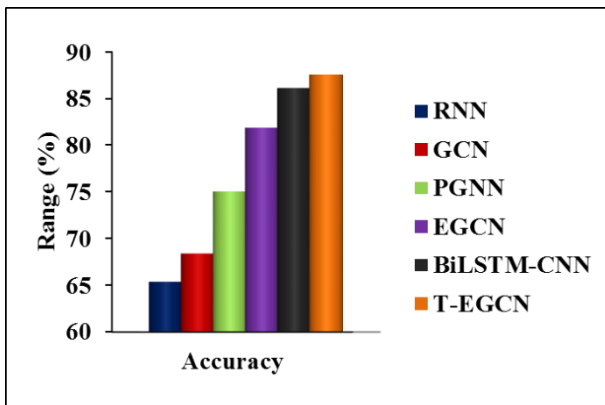


Figure 3. Comparison of accuracy

Figure 3 depicts the accuracy (in %) achieved by different models to identify rumor tweets. It indicates that the accuracy of T-EGCN is 33.92% greater than the RNN, 27.99% greater than the GCN, 16.68% greater than the PGNN, 6.95% greater than the EGCN, 1.67% greater than the BiLSTM-CNN models for identifying rumor tweets. This is because of learning the word vectors using an unsupervised GloVe algorithm rather than the Word2Vec to identify rumor tweets from the large Twitter corpora. Thus, it realizes that the T-EGCN model can

increase the accuracy of identifying rumors using different attribute vectors extracted from the large Twitter database.

4.4 Precision

It is the fraction of rumor tweets that are properly identified to the total identified tweets in a rumor label.

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

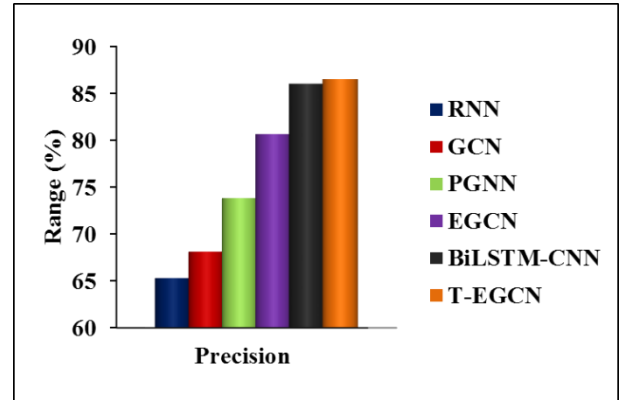


Figure 4. Comparison of precision

Figure 4 portrays the precision (in %) achieved by different models to identify rumor tweets. It notices that the precision of T-EGCN is 32.45% larger than the RNN, 26.99% larger than the GCN, 17.16% larger than the PGNN, 7.27% larger than the EGCN, 0.57% larger than the BiLSTM-CNN models to identify rumor tweets. This is attained due to the utilization of an unsupervised word-embedding algorithm for the large Twitter corpora during rumor identification. So, it proves that the T-EGCN model maximizes the precision of identifying rumors from the large Twitter database.

4.5 Recall

It determines the percentage of rumor tweets that are properly identified.

$$Recall = \frac{TP}{TP + FN} \quad (9)$$

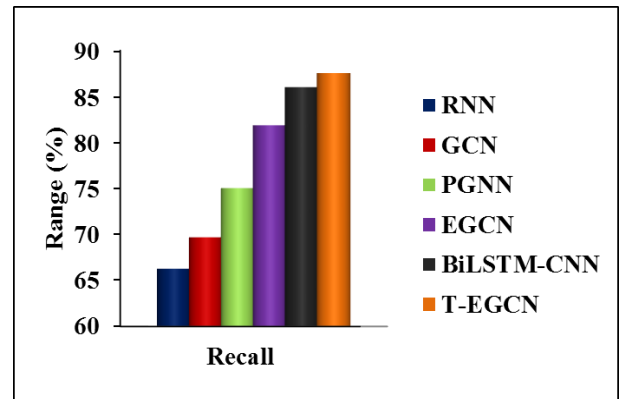


Figure 5. Comparison of recall

Figure 5 illustrates the recall (in %) achieved by different models to identify rumor tweets. It notices that the recall of T-

EGCN is 32.06% larger than the RNN, 25.6% larger than the GCN, 16.66% larger than the PGNN, 6.93% larger than the EGCN, 1.78% larger than the BiLSTM-CNN models to identify rumor tweets. Therefore, it exhibits that the T-EGCN model improves the recall of identifying rumors from the large Twitter database by using the unsupervised word-embedding scheme to learn the word vectors.

4.6 F-measure

It is the harmonic average between precision and recall.

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (10)$$

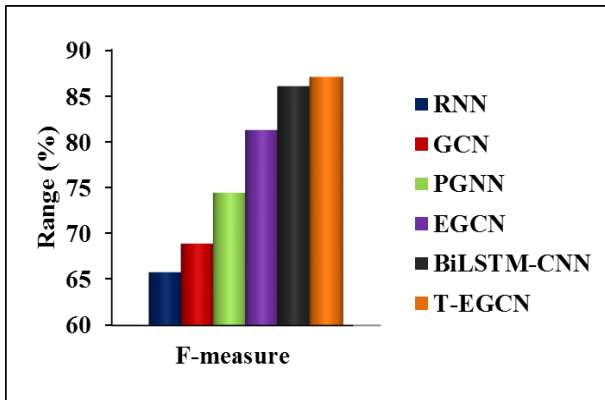


Figure 6. Comparison of f-measure

Figure 6 portrays the f-measure (in %) achieved by different models to identify rumor tweets. It notices that the f-measure of T-EGCN is 32.25% higher than the RNN, 26.28% higher than the GCN, 16.91% higher than the PGNN, 7.1% higher than the EGCN, 1.17% higher than the BiLSTM-CNN models to identify rumor tweets. This is attained due to the utilization of an unsupervised word-embedding algorithm for the large Twitter corpora during rumor identification. So, it proves that the T-EGCN model maximizes the f-measure of identifying rumors from the large Twitter database.

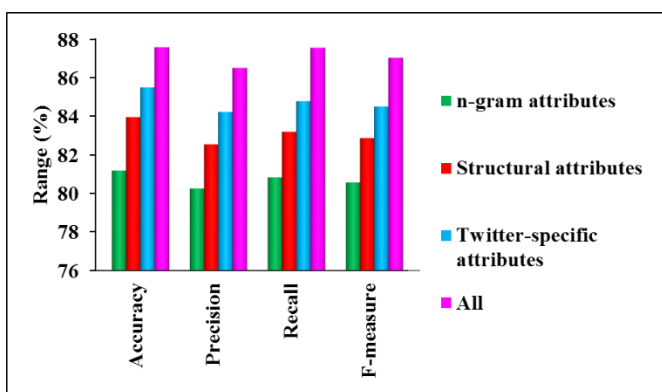


Figure 7. Comparing performance of T-EGCN model for tweet attributes vectors

Figure 7 depicts the efficiency of using multiple attributes in the T-EGCN model for rumor detection. Compared to the learning of attributes independently, a unified rumor attribute vector of tweets in the T-EGCN model enhances the detection of rumors efficiently. This results in better accuracy, precision, recall, and f-measure for the T-EGCN model compared to the

learning of independent attributes. This indicates the importance of all attributes related to the tweets for rumor detection.

5. CONCLUSION

In this study, the T-EGCN model was developed, which uses unsupervised learning-based word embedding to identify rumors from massive Twitter databases. The latent contextual semantic correlation and co-occurrence statistical variables among terms in tweets were used in this model. The word embeddings were then concatenated with the GloVe model's word attribute vectors, Twitter-specific attributes and n-gram attributes to form a rumor attribute vector of tweets. Further, this attribute vector was learned by the EGCN to recognize rumors from a huge Twitter corpus. At last, the experimental findings proved that the T-EGCN has an accuracy of 87.56% which is 16.18% higher than the other classical rumor identification models. On the other hand, a standard trait on Twitter is context attributes, i.e. a single post may be quite brief in a timeline, having relatively restricted context with variable time series lengths. So, future work will concentrate on solving this problem by developing a social-temporal sequence learning model which identifies rumors in the early phases of their progress.

REFERENCES

- [1] Paul, J., Parameswar, N., Sindhani, M., Dhir, S. (2021). Use of microblogging platform for digital communication in politics. *Journal of Business Research*, 127: 322-331. <https://doi.org/10.1016/j.jbusres.2021.01.046>
- [2] Cresci, S., Lillo, F., Regoli, D., Tardelli, S., Tesconi, M. (2019). Cashtag piggybacking: uncovering spam and bot activity in stock microblogs on Twitter. *ACM Transactions on the Web*, 13(2): 1-27. <https://doi.org/10.48550/arXiv.1804.04406>
- [3] Hamidian, S., Diab, M.T. (2019). Rumor detection and classification for twitter data. *arXiv preprint arXiv:1912.08926*, 1-7. <https://doi.org/10.48550/arXiv.1912.08926>
- [4] Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., Liu, Y. (2019). Combating fake news: a survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology*, 10(3): 1-42. <https://doi.org/10.48550/arXiv.1901.06437>
- [5] Pierri, F., Ceri, S. (2019). False news on social media: a data-driven survey. *ACM Sigmod Record*, 48(2): 18-27. <https://doi.org/10.1145/3377330.3377334>
- [6] Jung, A.K., Ross, B., Stieglitz, S. (2020). Caution: Rumors ahead – a case study on the debunking of false information on Twitter. *Big Data & Society*, 7(2): 1-15. <https://doi.org/10.1177/2053951720980127>
- [7] Habib, A., Asghar, M.Z., Khan, A., Habib, A., Khan, A. (2019). False information detection in online content and its role in decision making: a systematic literature review. *Social Network Analysis and Mining*, 9(1): 1-20. <https://doi.org/10.1007/s13278-019-0595-5>
- [8] Pathak, A.R., Mahajan, A., Singh, K., Patil, A., Nair, A. (2020). Analysis of techniques for rumor detection in social media. *Procedia Computer Science*, 167: 2286-

2296. <https://doi.org/10.1016/j.procs.2020.03.281>
- [9] Zhang, X., Ghorbani, A.A. (2020). An overview of online fake news: characterization, detection, and discussion. *Information Processing & Management*, 57(2): 1-26. <https://doi.org/10.1016/j.ipm.2019.03.004>
- [10] kumari Mukiri, R., Babu, B.V. (2021). Prediction of rumour source identification through spam detection on social Networks- a survey. *Materials Today: Proceedings*, 1-5. <https://doi.org/10.1016/j.matpr.2021.03.367>
- [11] Bondielli, A., Marcelloni, F. (2019). A survey on fake news and rumour detection techniques. *Information Sciences*, 497: 38-55. <https://doi.org/10.1016/j.ins.2019.05.035>
- [12] Alzanin, S.M., Azmi, A.M. (2018). Detecting rumors in social media: a survey. *Procedia computer science*, 142: 294-300. <https://doi.org/10.1016/j.procs.2018.10.495>
- [13] Rao, S., Verma, A.K., Bhatia, T. (2021). A review on social spam detection: challenges, open issues, and future directions. *Expert Systems with Applications*, 186: 1-31. <https://doi.org/10.1016/j.eswa.2021.115742>
- [14] Kumar, A., Singh, V., Ali, T., Pal, S., Singh, J. (2020). Empirical evaluation of shallow and deep classifiers for rumor detection. In *Advances in Computing and Intelligent Systems*, Springer, Singapore, 239-252. https://doi.org/10.1007/978-981-15-0222-4_21
- [15] Gongane, V.U., Munot, M.V., Anuse, A. (2022). Machine learning approaches for rumor detection on social media platforms: a comprehensive survey. *Advanced Machine Intelligence and Signal Processing*, 649-663. https://doi.org/10.1007/978-981-19-0840-8_50
- [16] Wang, S., Li, Z., Wang, Y., Zhang, Q. (2019). Machine learning methods to predict social media disaster rumor refuters. *International Journal of Environmental Research and Public Health*, 16(8): 1-16. <https://doi.org/16.10.3390/ijerph16081452>
- [17] Amoudi, G., Albalawi, R., Baothman, F., Jamal, A., Alghamdi, H., Alhothali, A. (2022). Arabic rumor detection: a comparative study. *Alexandria Engineering Journal*, 61(12): 12511-12523. <https://doi.org/10.1016/j.aej.2022.05.029>
- [18] Meel, P., Vishwakarma, D.K. (2020). Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Systems with Applications*, 153: 1-35. <https://doi.org/10.1016/j.eswa.2019.112986>
- [19] Bai, N., Meng, F., Rui, X., Wang, Z. (2021). Rumour detection based on graph convolutional neural net. *IEEE Access*, 9: 21686-21693. <https://doi.org/10.1109/ACCESS.2021.3050563>
- [20] Alzanin, S.M., Azmi, A.M. (2019). Rumor detection in Arabic tweets using semi-supervised and unsupervised expectation-maximization. *Knowledge-Based Systems*, 185: 1-9. <https://doi.org/10.1016/j.knosys.2019.104945>
- [21] Fard, A.E., Mohammadi, M., Chen, Y., Van de Walle, B. (2019). Computational rumor detection without non-rumor: a one-class classification approach. *IEEE Transactions on Computational Social Systems*, 6(5): 830-846. <https://doi.org/10.1109/TCSS.2019.2931186>
- [22] Wu, Z., Pi, D., Chen, J., Xie, M., Cao, J. (2020). Rumor detection based on propagation graph neural network with attention mechanism. *Expert Systems with Applications*, 158: 1-31. <https://doi.org/10.1016/j.eswa.2020.113595>
- [23] Alkhodair, S.A., Ding, S.H., Fung, B.C. Liu, J. (2020). Detecting breaking news rumors of emerging topics in social media. *Information Processing Management*, 57(2): 1-13. <https://doi.org/10.1016/j.ipm.2019.02.016>
- [24] Kotteti, C.M.M., Dong, X., Qian, L. (2020). Ensemble deep learning on time-series representation of tweets for rumor detection in social media. *Applied Sciences*, 10(21): 1-21. <https://doi.org/10.48550/arXiv.2004.12500>
- [25] Yu, K., Jiang, H., Li, T., Han, S., Wu, X. (2020). Data fusion oriented graph convolution network model for rumor detection. *IEEE Transactions on Network and Service Management*, 17(4): 2171-2181. <https://doi.org/10.1109/TNSM.2020.3033996>
- [26] Song, C., Yang, C., Chen, H., Tu, C., Liu, Z., Sun, M. (2021). CED: Credible early detection of social media rumors. *IEEE Transactions on Knowledge and Data Engineering*, 33(8): 3035-3047. <https://doi.org/10.1109/TKDE.2019.2961675>
- [27] Chen, X., Zhou, F., Zhang, F., Bonsangue, M. (2021). Catch me if you can: A participant-level rumor detection framework via fine-grained user representation learning. *Information Processing & Management*, 58(5): 1-19. <https://doi.org/10.1016/j.ipm.2021.102678>
- [28] 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(4): 4315-4333. <https://doi.org/10.1007/s12652-019-01527-4>
- [29] Choudhary, A., Arora, A. (2021). Linguistic feature based learning model for fake news detection and classification. *Expert Systems with Applications*, 169: 1-25. <https://doi.org/10.1016/j.eswa.2020.114171>
- [30] Yao, L., Mao, C., Luo, Y. (2019). Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(1): 7370-7377.
- [31] https://figshare.com/articles/dataset/PHEME_dataset_for_Rumor_Detection_and_Veracity_Classification/6392078