Deep Learning-Based Rumor Detection on Social Media

Bevin Biju Thomas and Cinu C. Kiliroor

Abstract Rumor detection is a highly prominent issue especially during this pandemic that requires adequate attention and addressal. Citizens confined to their homes rely on the use of smartphones, and other devices by means of social media and online news publishing websites so as to gain thorough insights into various topics of public interest. However, due to this very convenience, since there exists a large plethora of articles published, the human judgment and capability to distinguish news to be true or false is often clouded due to the increasing consumption. Thus, there arises a need to separate and distinguish news articles which are true from the articles that circulate false news that often cause confusion and unnecessary panic with rumors being circulated around.

Keywords Rumor detection · Deep learning · Bi-LSTM

1 Introduction

Due to the rise in the use of Social Media, and the introduction of various convenient portable devices which provide access to the Internet, the accessibility to knowledge has never been easier. The enormous availability of various social media applications and news applications has made a plethora of information and knowledge accessible within one's fingertips. This very convenience and access has not only made human lives easier but has also made it difficult to discern rumors.

Existing methods utilize Machine Learning methods with limited features for training the data. A realistic measure can be achieved by utilizing more features such as sentiment and other user-based features. This is done because these features help in understanding user behavior and its correlation to rumors. The project utilizes Deep Learning-based methodologies for enhancing the dynamic ability of the model

B. B. Thomas · C. C. Kiliroor (✉)
Vellore Institute of Technology, Chennai, India
e-mail: cinu.kiliroor@vit.ac.in

B. B. Thomas
e-mail: bevinbiju.thomas2017@vitstudent.ac.in

P. K. Mallick et al. (eds.), *Electronic Systems and Intelligent Computing*, Lecture Notes in Electrical Engineering 860, https://doi.org/10.1007/978-981-16-9488-2_13

to distinguish rumors and non-rumors. It also utilizes the incorporation of word embedding vectors.

2 Background

2.1 Literature Study

Jain et al. [1] utilized Support Vector Machines. The article extraction is done by means of retrieving the title from RSS feeds, the features involving author, title and content are extracted and then the validation is done by applying the trained classifier on the given article and also verified with the user input gathered from browsing if the article exists in multiple websites or not. The model's performance resulted in up to 93.50% accuracy.

In [2], they define a certain threshold which utilizes the number of likes feature and if the number of likes feature contains a value less than the threshold, it makes use of a content-based classifier and if greater or equal to the threshold, it utilizes a social interaction-based classifier which makes use of Logistic Regression and Harmonic Boolean label crowdsourcing on Social Signals.

The approach utilized by [3] involves the collection of news stories from legitimate news websites and the same is done for Facebook posts through GraphAPI; stories and articles are further processed to determine if there exists any content similarity between both; the comments and replies for each post are utilized for sentiment analysis using VADER. This is then followed by the use of ML algorithms and then compared based on accuracy, precision and recall.

The comparison in [4] showed that Deep Learning methods performed better in accuracy, precision and recall.

In [5], a weightage factor was assigned to rumors because rumors are a minority class; this is done by assigning this weighing factor to terms that frequently occur within the rumors class, while also utilizing tf-idf scores and adding this score to the standard tf-idf score.

The study in [6] compares existing Machine Learning approaches. The approaches considered are JRip, ZeroR, Naive Bayes, Random Forest, OneR and Hoeffding Tree.

The approach in [7] utilizes a methodology where the rumor label is not treated as a binary classification rather it is classified into 4 classes, namely true, partially true, partially false and false. The approaches utilized here are, namely K-Means Clustering and Support Vector Machines.

The study in [8] considers various types of features that can be considered. The features were categorized into Content-based features and Context-based features. Each of them was further classified into Lexical, Syntactic and Semantic under Content-based features followed by User-based and Network-based under Context-based features.

Hybrid deep learning approaches are utilized in [11]. Here, a CNN-RNN hybrid-based approach was utilized in the application of rumor detection.

2.2 Existing Features

Utilization of user-based features such as Users that have a bio, Users with a display picture, Whether User is a verified user or not, average number of followers, average number of posts posted by users, duration of user's account and identification of how prominent the user's account is by considering followers: following ratio and average number of retweets.

Content-based features are character count, number of stop words used, grammatical-based features through POS tagging through Natural Language Processing, considering if there exist question marks and exclamations, average length of posts for a particular topic or event, sentiment score of posts, calculating overall sentiment of a post's comments or thread and time series data.

3 Methodology

3.1 Overview

A hybrid deep learning approach is adopted with the addition of more features for higher accuracy and capability to detect rumors. This approach is compared with individual algorithms' performance as well as with Machine Learning algorithms. The addition of word embedding vectors is considered so as to represent various words as vectors. This is done in order to determine whether the existence of certain words with higher frequency within the rumors class possesses a correlation to it being a rumor or a non-rumor.

The system will utilize one of the hybrid deep learning approaches Bi-LSTM with the use of word embedding vector - GloVe. This is then compared with existing machine learning classifiers and the individual performance of Bi-LSTM and LSTM. The training data for the machine learning classifiers are mostly user- and textual content-based data which are derived during the pre-processing phase. And for the Bi-LSTM with word embedding, Bi-LSTM and LSTM, the training data input is the textual content of the tweet.

3.2 Logistic Regression

Logistic regression utilizes a hypothesis that limits the cost function within the range of [0,1]. Linear functions fail to represent this due to their likelihood of having values beyond this range.

$$0 \leq h_\theta \leq 1 \tag{1}$$

Logistic Regression is very efficient at classifying unknown records, and can interpret model coeffcients as indicators of feature importance. However, the major drawback with Logistic Regression is the assumption of linearity between the dependent and independent variables. Moreover, its usage is restricted to only predicting discrete functions.

3.3 Decision Tree

The Decision Tree algorithm involves splitting the data continuously according to certain criteria until a tree representation is obtained. The tree consists of 2 entities, namely decision nodes and leaf nodes, where the decision nodes represent the criteria to split the data and the leaf nodes represent possible outputs that can be derived based on the criteria. The criteria involved are often based on attributes values in the dataset. This splitting operation is done until the leaf node can be determined for each attribute based on the decision criteria.

3.4 Random Forest

Random Forest consists of a large number of individual decision trees that operate as an ensemble (a group of trees). Each individual tree splits out into a class prediction and the class which possesses the most votes becomes the model's prediction. While some trees may vary inaccurately, the rest of the numerous tree predictions will be right, so as a result, the group of trees is able to proceed in the right order neglecting the inaccurate ones. Thus, its capability to have multiple decision trees and then combine them in order to get a more accurate prediction makes it ideal for both classification and regression problems.

3.5 Long Short-Term Memory Networks

LSTM consists of 3 gates, namely input gate, forget gate and output gate. Within the LSTM, the cell state carries information in a sequential manner and acts as the memory unit of the network which makes it capable of carrying relevant information needed throughout the processing of an entire sequential data, thus, data present in earlier processing steps can be used in the later steps as well. The gates are neural network units capable of determining which information is relevant and can be stored in the cell state. The gates' functioning is based on sigmoid function and is referred to within LSTM as sigmoid activations. The range of values varies between 0 and 1.

The forget gate determines the information that is to be carried over. The data from the preceding hidden state and the data repressing the input is passed through the sigmoid function; the values lie within the range of [0,1]; values that are immediate to 0 can be discarded and values immediate to 1 are retained. Input gate is utilized so as to update the cell state. The input and data from the prior hidden state are passed directly to the sigmoid function which determines the values to be updated by transforming their values within the range [0,1]. This is then followed by passing the hidden state and input through the tanh function which transforms the values between -1 and 1. This is then followed by the product of values from the sigmoid function and the tanh function, followed by the sigmoid function's value determining which information is to be retained from the tanh value. The cell state involves the calculation of the dot product between the forget vector and the cell state. This is done in order to drop values present in the cell state that are immediate to 0. Pointwise addition is performed on the output of the input gate which updates the cell state to newer values that the neural net can find pertinent. The output gate determines the next hidden state. The prior hidden state along with the input is passed into a sigmoid function. Then the resultant cell state is passed to the tanh function. The product of the tanh function value and the sigmoid function value determine which information the hidden state should carry over. The output is the hidden state. The new cell state and the new hidden state are then carried over to the next step. The input gate decides the information which is relevant to add from the current step. The output gate then decides the next hidden state.

3.6 Bi-directional Long Short-Term Memory Networks

Various Hybrid Deep Learning approaches other than Bi-LSTM include ConvLSTM which integrates both CNN and LSTM architectures and CNN-RNN which utilizes both CNN and RNN. Hybrid Deep Learning was adopted to improve the output of sequential data. Bi-LSTM utilizes a sequence processing model that comprises 2 LSTMs where one LSTM passes the input in a forward direction and the other in a backward direction and then concatenates both the results for every processing step. This effectively increases the amount of relevant information that can be considered which helps in increasing the context availability and understanding. This makes the model capable of predicting which words follow and precede a particular word for a given sentence. The addition of an LSTM that runs backward helps preserve information by utilizing the combination of 2 hidden states which are capable of preserving data from both forward and backward directions.

3.7 Word Embedding Vectors

GloVe is capable of procuring vector representations for a given textual data input. The importance and usage of word embeddings are that they can be utilized for grouping words that are synonyms to each other in context and meaning in a similar representation. Each word is mapped to one word embedding vector and their distributed representation is contingent on the usage of words. Thus, due to this, the words that are utilized in similar sentences will be stored in vector form with similar representation.

GloVe's approach to word embedding representation is based on the co-occurrences or the frequency of a word in the entire document. The embeddings of GloVe represent numeric probabilities of whether a certain word and another word can appear together or not. Since the predictive modeling aspect is taken care of by utilizing Bi-directional LSTM, GloVe word embeddings will be utilized.

The pre-processing step in Fig. 1 involves the conversion of rumor events data in json format to csv. More features were also introduced and added along with the existing features during the conversion from json to csv format.

This is then followed by the EDA phase involved in analyzing the distribution of data in order to ensure if there exists a fair distribution between rumors and non-rumors within the overall data as well as for each and every event in the training dataset. The inference can be drawn from the data by visualizing them.

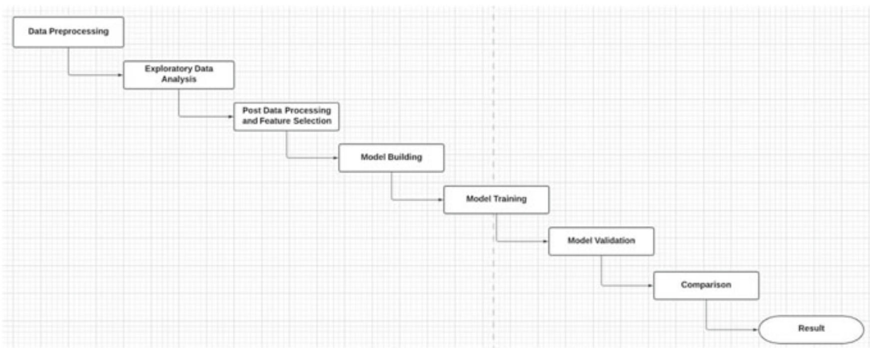

Fig. 1 Rumor detection workflow diagram

In the post-data processing and feature selection, the attributes that are not necessary are dropped and label encoding is applied for those features that are in textual format.

The model building phase involves identifying algorithms that will be effective for the application and implementing a few ML algorithms.

The model validation involves generating a classification report where metrics are generated in order to compare the algorithms' performance and finding which performed the best among them.

For both LSTM and Bi-LSTM in order to implement and compare the performance of them individually with Bi-LSTM utilizing word embedding vectors, the textual content of the tweet is utilized as the feature in the post-data processing stage and within that stage, pre-processing techniques are applied on the tweet's text.

From Fig. 2, the textual input is converted into word embedding vectors and then utilizing GloVe word embedding, it is passed to the Bi-LSTM model for training. The training occurs in such a way that one of the LSTMs passes the input vectors

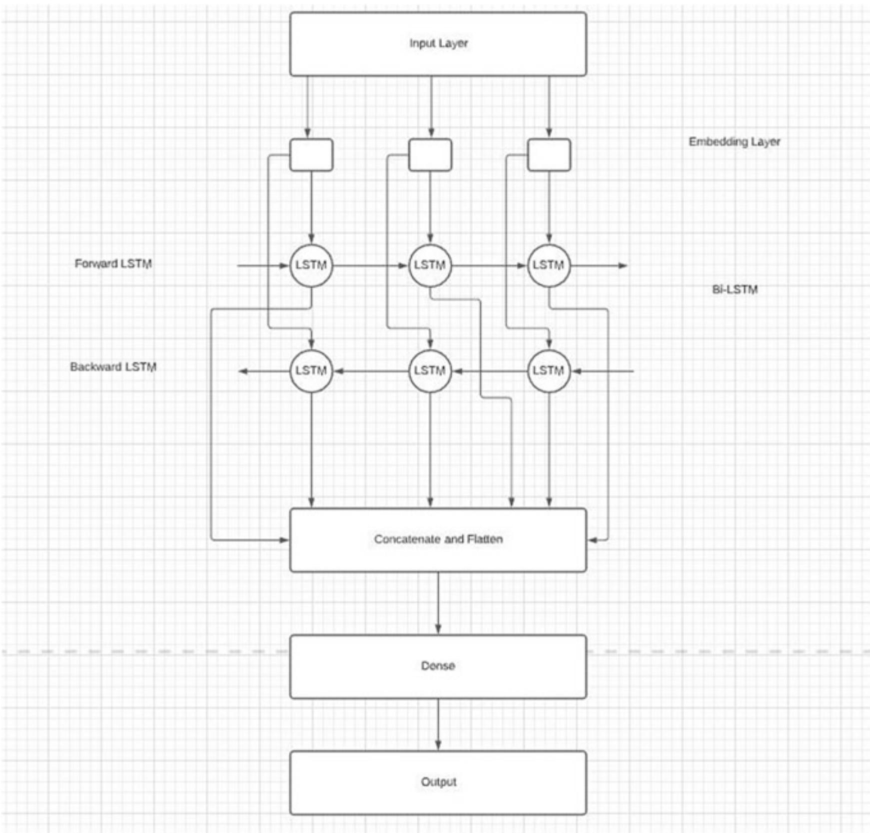

Fig. 2 Architecture diagram

in a forward direction and the other in a backward direction. The resultant vectors are then combined in the concatenate and flatten step. This is to increase the context availability. Thus, the model's capability of predicting words immediately follows a particular word present as a vector and the word that precedes it.

4 Experimental Setting

4.1 Data

Pheme-rnr-dataset was used. The dataset comprises Twitter data from 5 real-life events, namely Ottawa shooting, Charlie Hebdo shooting, Ferguson unrest, Sydney siege, also known as Lindt Cafe Siege and German Wings Flight 9525 crash. The dataset contains Twitter conversation threads for these events. Each event contains its own directory with 2 separate folders containing the tweets categorized as Rumors and Non-Rumors.

5 Experimental Results and Analysis

From Table 1, it can be inferred based on the F-1 score that Random Forest had the best performance among the 3 machine learning classifiers. From Table 2, based on the accuracy, Random Forest again was the most effective. After the model training for both LSTM and Bi-LSTM and considering the first epoch training accuracy to compare them, it is observed that utilizing a hybrid deep learning approach has a

Table 1 Machine learning algorithms comparison

Algorithm	F-1 Score	Precision	Recall
Logistic regression	0.78	0.64	1.00
Decision tree	0.76	0.76	0.76
Random forest	0.83	0.76	0.91

Table 2 Performance comparison of algorithms

Algorithm	Accuracy(%)
Logistic regression	63.71
Decision tree	69.92
Random forest	75.52
LSTM	86
Bi-LSTM	90
Bi-LSTM with GloVe	85

slightly better performance compared to the individual unidirectional approach. With GloVe, it is slightly lesser but close to the LSTM'S and Bi-LSTM's accuracy. The utilization of deep learning approaches possesses higher accuracy compared to that of machine learning approaches. This makes deep learning approaches superior in their capability to detect rumors to that of machine learning approaches.

The result has shown an increase in the accuracy when deep learning approaches were utilized. This is possibly due to the higher context availability present due to the utilization of the hybrid deep learning approach and word embedding vectors. The selection of Bi-LSTM among other hybrid deep learning approaches was due to the capability of having the input fed in both forward and backward directions which increases the context understanding and availability as the model will be capable of predicting input forward and backward. Having tested out the machine learning classifier approaches and the individual performances of LSTM having also shown that Bi-directional LSTM is more beneficial than unidirectional LSTM, hence, the utilization of hybrid deep learning approaches can significantly enhance the context availability needed in the application of detecting of rumors.

6 Conclusion

Since the ability to attain contextual understanding is higher in the case of deep learning approaches, the accuracy in predicting and distinguishing rumors is higher. The utilization of word embedding vector GloVe performed better than machine learning classifiers and was close in terms of accuracy to the individual deep learning approaches of unidirectional LSTM and Bi-directional LSTM.

References

1. Jain A, Shakya A, Khatter H, Kumar Gupta A (2019) A smart system for fake news detection using machine learning. In: 2019 2nd international conference on issues and challenges in intelligent computing techniques
2. Marco L. Della Vedova, Eugenio Tacchini, Stefano Moret, Gabriele Ballarin, Massimo DiPierro, Luca de Alfaro (2018) Automatic online fake news detection combining content and social signals. In: 2018 22nd conference of open innovations association (FRUCT) (2018)
3. May Me Me Hlaing, Nang Saing Moon Kham.: Dening News Authenticity on Social Media Using Machine Learning Approach, 2020 IEEE Conference on Computer Applications(ICCA) (2020)
4. Wenlin Han, Varshil Mehta.: Fake News Detection in Social Networks Using Machine Learning and Deep Learning: Performance Evaluation, 2019 IEEE International Conference on Industrial Internet (2019)
5. Uddipta Bhattacharjee, Srijith P.K, Mauendra Sankar Desarkar.: Term Speci c TF-IDF Boosting for Detection of Rumors in Social Networks, 2019 11th International Conference on Communication Systems and Networks (2019)
6. Harun Bingol, Bilal Alatas (2019) Rumor Detection in Social Media Using Machine Learning Methods, International Informatics and Software Engineering Conference

7. Faraz Ahmad, Syed Afzal Murtaza Rizvi.: Information Credibility on Twitter Using Machine Learning Techniques, International conference on Futuristic Trends in Networks and Computing Technologies (2019), pp 371–381
8. Bondielli A, Marcelloni F (2019) A survey on fake news and rumor detection techniques. Inf Sci 497:38–55 (2019)
9. Data Source. https://gshare.com/articles/dataset/PHEMEdatasetofrumoursandnon-rumours/4010619
10. Glove Pre-trained word embedding vector. https://nlp.stanford.edu/projects/glove/
11. Jamal Abdul Nasir, Osama Subhani Khan, Iraklis Varlamis (2020) Fake news detection: a hybrid CNN-RNN based deep learning approach. Int J Inf Manag Data Insights 1(1)