Information Systems Frontiers
https://doi.org/10.1007/s10796-020-10040-5

Attention-Based LSTM Network for Rumor Veracity Estimation of Tweets

Jyoti Prakash Singh[1] · Abhinav Kumar[1] · Nripendra P. Rana[2] · Yogesh K. Dwivedi[3]

Abstract

Twitter has become a fertile place for rumors, as information can spread to a large number of people immediately. Rumors can mislead public opinion, weaken social order, decrease the legitimacy of government, and lead to a significant threat to social stability. Therefore, timely detection and debunking rumor are urgently needed. In this work, we proposed an Attention-based Long-Short Term Memory (LSTM) network that uses tweet text with thirteen different linguistic and user features to distinguish rumor and non-rumor tweets. The performance of the proposed Attention-based LSTM model is compared with several conventional machine and deep learning models. The proposed Attention-based LSTM model achieved an F_1-score of 0.88 in classifying rumor and non-rumor tweets, which is better than the state-of-the-art results. The proposed system can reduce the impact of rumors on society and weaken the loss of life, money, and build the firm trust of users with social media platforms.

Keywords Rumor · Twitter · Deep learning · Machine learning

1 Introduction

Online social media like Twitter and Facebook has become an inescapable part of everyday life (Dwivedi et al. 2015; Kumar and Rathore 2016; Alalwan et al. 2017; Alryalat et al. 2017; Tamilmani et al. 2018; Shareef et al. 2019). Twitter is currently one of the most preferred online social media platforms for users to share information in the form of short text messages limited to 280 characters termed as tweets. Users read and forward it to another group (retweet) quickly compared to other social media platforms. Therefore, information spreads rapidly through the network of users. Several breaking news is reported to first appeared on Twitter before being circulated through traditional news media (Singh et al. 2019a). Twitter data has been effectively used in disaster management (Singh et al. 2019a; Kumar et al. 2017; Kumar and Singh 2019; Kumar et al. 2020; Abedin and Babar 2018; Ghosh et al. 2018), location prediction (Kumar and Singh 2019), and customer relationship management (Kapoor et al. 2018; Kizgin et al. 2018; Baabdullah et al. 2018; Shareef et al. 2019), antisocial activities tracking (Oh et al. 2011), government policies monitoring (Singh et al. 2019c), traffic monitoring (Vallejos et al. 2020) to name a few.

Twitter does not have any high-level filtering or moderation mechanism to validate the authenticity of the posted contents which result in the spread of rumor (Ma et al. 2016; Mondal et al. 2018; Singh et al. 2019b), spamming (Aswani et al. 2018), sentiment bias (Smith et al. 2018) and other unsocial behaviors. "Rumors are unverified and instrumentally relevant information in circulation that arises in contexts of ambiguity, danger, or potential threat" (DiFonzo and Bordia 2007). The open nature of Twitter is an appropriate place for rumor makers to post and spread rumors.

✉ Nripendra P. Rana
nrananp@gmail.com

Jyoti Prakash Singh
jps@nitp.ac.in

Abhinav Kumar
abhinavanand05@gmail.com

Yogesh K. Dwivedi
ykdwivedi@gmail.com

[1] National Institute of Technology Patna, Patna, India

[2] School of Management, University of Bradford,
Richmond Road, Bradford BD7 1DP, UK

[3] Emerging Markets Research Centre (EMaRC),
School of Management, Swansea University,
Bay Campus, Swansea, UK

Published online: 12 August 2020

The spread of rumors can have severe impacts on society as rumors mislead public opinion, weaken social order, diminish the trust of citizens in government, decrease the legitimacy of the government, and lead to a significant threat to social stability (Huang 2017; Liang et al. 2015; Khan and Idris 2019; Lee et al. 2015). For example, on 23 April 2013, the rumor "An attack on the White House" was posted from the Associated Press hacked account, resulting in a loss of 136 billion dollars in the stock market within a few seconds of the report (Liu et al. 2019). On August 25, 2015, the rumor "shootouts and kidnappings by drug gangs happening near schools in Veracruz" propagated via Twitter and Facebook, created severe chaos in the city, causing 26 car crashes as people left their vehicles in the middle of the highway and raced to pick up their kids from school (Ma et al. 2016). In September 2019, during heavy rain in the Patna city, India, a rumor "Now crocodiles are floating from Ganga river to residential areas of Patna #patnafloods" caused a lot of fear and panic across people. False news over Facebook during the 2016 U.S. presidential election influenced people's choice of the vote and had a significant impact on the election results (Allcott and Gentzkow 2017; Meel and Vishwakarma 2019). To increase the reliability of online social networks and mitigate the devastating impacts of false and rumorous information, timely detection, and containment of rumor content circulating on social media is essential. An automated rumor detection system can debunk rumors at an early stage to limit the spread of rumors and mitigate their harmful effects (Ma et al. 2016; Meel and Vishwakarma 2019; Singh et al. 2019b). However, the identification of false and rumored information disseminated through social media is a challenging research task (Lozano et al. 2020; Meel and Vishwakarma 2019; Serrano et al. 2015; Singh et al. 2019b).

Kim et al. (2019) suggested assigning a rating value to the source could be a viable measure against fake news. They also detailed the rating mechanism for the news sources. They further reported that a low source rating highly affects the believability for unknown sources (Kim and Dennis 2019). Kwon et al. (2017) found by statistical techniques that the over a long term window structural and temporal features can distinguish rumors from non-rumors but these features are unavailable at the initial phase of the rumor propagation. Hence, they suggested using user and linguistic features for the early detection of rumors. Ma et al. (2015) used a wide range of manually crafted features based on textual content, user, and diffusion path of tweets to classify a tweet as rumor and non-rumor. Along with linguistic features, tweet characteristics such as whether a tweet supports the rumor, denies a rumor, questions a rumor, or is a regular post, is also used to identify rumor and non-rumor tweets by researchers (Derczynski et al. 2017; Enayet and El-Beltagy 2017). Chen et al. (2018a) used a deep learning model with an attention mechanism to extract textual features from tweets to detect rumors tweets.

In this article, we extended the feature set to identify a rumorous tweet by extracting textual features using the deep learning model and thirteen other linguistic and user features from a tweet to create a hybrid feature set. Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) network were used as deep learning models for automatic feature extraction from tweet texts. Further, to find the best features from the created hybrid feature set, the Particle Swarm Optimization (PSO) algorithm is used. PSO is a population-based optimization algorithm that selects the best performing subset of features from the created hybrid feature set to yield an optimal feature set. The optimal feature set is then used to classify tweets into rumor and non-rumor classes using seven different machine learning classifiers (i) Support Vector Machine (SVM), (ii) Random Forest, (iii) Logistic Regression, (iv) Naive Bayes, (v) K-Nearest Neighbour (KNN), (vi) Gradient Boosting, and (vii) Decision Tree. An attentional based LSTM network is also proposed using this hybrid feature to classify tweets into rumor and non-rumor classes. The contributions of the proposed work can be summarized as follows:

- Creating a hybrid feature set from tweets to classify them into rumor or non-rumor class. One group of the features is extracted automatically from tweet text by deep learning models while another group is extracted manually from tweet text and user characteristics.
- Selecting an optimal feature set through the PSO algorithm for further classification.
- Proposing attention based Long-Short Term Memory network with the hybrid feature set to classify tweets into rumor or non-rumor class.
- Comparing the performance of the attention-based LSTM network with the mentioned seven different machine learning classifiers.

The rest of the article is organized as follows: Section 2 deals with the related works; Section 3 describes the detailed methods used; Section 4 listed various experimental results. The outcome of the experiments is discussed in Section 5. Finally, we concluded the paper in Section 6.

2 Literature Review

The information authenticity determination of social media content is a complicated task. One line of research has focused on extracting the relevant features from the social media post using the machine and deep learning techniques to identify rumors, while the other line of research has concentrated on the users who spread rumors across the network. In this section, we discuss a brief description of some of the potential works proposed in this domain.

Castillo et al. (2011) extracted several features such as content-based, user-based, topic-based, and propagation-based features to build a classifier to classify microblog posts of trending topics as credible or not credible. To identify rumors, Qazvinian et al. (2011) used three different types of features, (i) content-based, (ii) network-based, and (iii) microblog-specific memes. Using those features, they identified the disinformers and users who support and further help spread the rumor. Liang et al. (2015) extracted eleven different linguistic and user features from the messages posted on Sina Weibo and used machine learning techniques to identify the rumor post. Zhao et al. (2015) used a different set of regular expressions to identify the inquiry related tweets and clustered similar posts. Then they gathered tweets that did not contain the inquiry terms, and finally, ranked the clusters by their likelihood of containing disputed factual claims to identify the rumors. Zubiaga et al. (2016) analyzed how users spread, support, or deny the rumors related posts. Their study suggested that there is a need to develop robust machine learning-based models to provide real-time assistance in finding the veracity of the rumors. Lukasik et al. (2016) used temporal and textual information from the tweets and applied Hawkes Processes to model the rumor stance classification on Twitter datasets. Hamidian and Diab (2016) used Tweet Latent Vector (TLV) features by applying Semantic Textual Similarity (STS), which generates a 100-dimensional feature vector for each tweet to retrieve rumor-related tweets. Oh et al. (2018) performed extensive studies on the acceptance of rumor and its consequences during crises. They found that people with closer ties were likely to believe the rumors as fact. Mondal et al. (2018) proposed a technique to detect rumor at the early stage in the aftermath of a disaster. They used the probabilistic model by incorporating prominent rumor propagation characteristics from the 2015 Chennai flood.

The idea of finding the source node on the network spreading the rumor was addressed by Jain et al. (2016). They proposed heuristic algorithms based on hitting time statistics of the surrogate random walking method to estimate the maximum likelihood of the source of the rumor. They tested their model on some standard and real-world networks. Their results outperformed many centrality-based heuristics that have traditionally been used in the identification of the rumor source. Ma et al. (2017) created the microblog post propagation trees to learn useful information about how the original message is transmitted and developed over time. Then a kernel-based propagation tree was used to capture the high-level patterns to separate rumors from the original microblog post. Srivastava et al. (2017) used combinations of statistical classifiers, hand-written patterns, and heuristics modules to perform both stance classification and veracity prediction of tweets. In their analysis, Maximum Entropy, Naive Bayes, and Winnow classifiers were used. Liu et al. (2017) investigated the rumor detection problem from a diffusion perspective and extracted content, user, temporal, and structural based features from Sina Weibo messages. They used these extracted features with SVM classifier to classify messages into rumor and non-rumor classes. The extensive survey of rumor detection techniques can be seen in Zubiaga et al. (2018) and Meel and Vishwakarma (2019).

Most of the previous approaches depend on the different features extracted from linguistic information. The performance of these systems depended heavily on how efficiently the features were extracted. Recently, some deep learning-based models (Ma et al. 2016; Chen et al. 2017; Ajao et al. 2018; Asghar et al. 2019; Chen et al. 2018a) have been proposed to reduce the limitations of handcrafted features to identify rumor messages. Ma et al. (2016) used a recurrent neural network (RNN) model to predict the veracity of the social media post by automatic feature learning and semantic information learning capability. Chen et al. (2017) used GloVe pre-trained word embedding to convert textual information into vector form and then applied a convolutional neural network to detect tweet stance and determine rumor veracity. Liu et al. (2019) captured the dynamic changes of forwarding contents, spreaders, and diffusion structure of the spreading process and then applied the LSTM network to identify rumors. Chen et al. (2018b) developed a model for learning the normal behavior of individual users using a recurrent neural network and autoencoder. Errors from different types of Weibo users have been used to evaluate whether it is a rumor or not using self-adapting thresholds. Rath et al. (2017) used a GRU-based RNN model to identify rumor spreaders using user embedding as input features generated by the believability re-weighted retweet network. Ajao et al. (2018) proposed a hybrid model combining long-term recurrent neural network and convolutional neural network (LSTM-CNN) models for the classification of tweets into rumor and non-rumor classes. They found that good accuracy can be achieved with deep neural network-based models in case of rumor detection, even with a small amount of training data. Asghar et al. (2019) proposed a deep neural network based on Bidirectional Long-Short Term Memory with Convolutional Neural Network (BiLSTM-CNN) to classify tweets into rumor and non-rumor classes. They have achieved a state-of-the-art result with the publicly available Pheme (Zubiaga et al. 2016) dataset. Some of the potential works related to rumor identification are summarized in Table 1.

The work done by Chen et al. (2018a) is close to our proposed work. They used the tf-idf (Sammut and Webb 2010) representation of the tweets in a matrix form. This matrix is used by the attention-based recurrent neural

Table 1 List of some of the potential works for the rumor identification

Authors	Task	Platform	Methodology	Features	Results
Castillo et al. (2011)	Twitter news credibility	Twitter	J48 decision tree	Content, user, topic, and propagation-based features	$F_1 - score = 0.86$
Qazvinian et al. (2011)	Rumor identification	Twitter	Belief classification	Content, network, and microblog-specific memes	F_1-score = 0.93
Liang et al. (2015)	Rumor identification	Sina Weibo	Various Classifiers	Linguistic and user features	F_1-score =0.55-0.86
Lukasik et al. (2016)	Rumor classification	Twitter	Hawkes Processes	Temporal and textual information	Accuracy = 67.77 - 72.93
Hamidian and Diab (2016)	Rumor identification	Twitter	SVM Tree Kernel	Tweet Latent Vector (TLV) feature	Precision = 0.97
Mondal et al. (2018)	Rumor identification	Twitter	Probabilistic model	Structural, linguistic, social ties	F_1-score = 0.67
Ma et al. (2017)	Rumor identification	Twitter	Propagation Tree Kernel	Tweet propagation structure	Accuracy = 0.75
Srivastava et al. (2017)	Stance classification and veracity prediction	Twitter	Entropy, Naive Bayes, and Winnow	Hand-written patterns and rules (heuristics)	Accuracy = 0.39-0.71
Liu et al. (2017)	Rumor identification	Sina Weibo	Hybrid SVM	Content, user, temporal, and structural based features	Accuracy = 0.94
Liu et al. (2019)	Rumor identification	Sina Weibo	Long Short-Term Memory (LSTM)	Word embedding	Accuracy = 0.95
Chen et al. (2018b)	Rumor identification	Sina Weibo	Recurrent Neural Networks and Autoencoders	Content and network-based features	F_1-score = 0.89
Rath et al. (2017)	Rumor identification	Twitter	GRU-based RNN model	User embedding	Accuracy = 0.70
Ma et al. (2016)	Rumor identification	Twitter & Sina Weibo	Recurrent Neural Networks (RNN)	Word embedding	F_1-score = 0.90-0.91
Chen et al. (2017)	Stance classification & veracity prediction	Twitter	Convolutional Neural Network (CNN)	Word embedding	Accuracy = 0.70
Ajao et al. (2018)	Rumor identification	Twitter	LSTM, LSTMDrop, LSTM-CNN	Word embedding	Accuracy = 0.74-0.82
Asghar et al. (2019)	Rumor identification	Twitter	BiLSTM-CNN	Word embedding	F_1-score = 0.86
Chen et al. (2018a)	Rumor identification	Twitter & Sina Weibo	Attention based LSTM	TF-IDF matrix	Precision = 0.72-0.74

network-based model to identify rumor tweets. In their work, they only used tweet text to identify rumors by neglecting several useful features. The role of user features is prominent, as discussed by several past works of literature (Liang et al. 2015; Zubiaga et al. 2016; Castillo et al. 2011). We are incorporating several user features with the tweet texts and proposing an Attention-based LSTM model that uses word embedding for the tweet text to better learn the semantics of the words to classify rumor and non-rumor tweets.

3 Methodology

We conducted extensive experiments with the conventional machine and deep learning models to identify rumor veracity. Seven different machine learning models were used: (i) Support Vector Machine (SVM), (ii) Random Forest (RF), (iii) Logistic Regression (LR), (iv) K-Nearest Neighbor (KNN), (v) Naive Bayes (NB), (vi) Gradient Boosting (GB), and (vii) Decision Tree (DT). In the case of deep learning, three different models are used: (i) Long-Short Term Memory (LSTM), (ii) Convolutional Neural Networks (CNN), and (iii) Attention-based Long-Short Term Memory. We also used Particle Swarm Optimization (PSO) with deep learning models to select the best performing set of features.

3.1 Data Description

The proposed methodology is validated with the publicly available Pheme (Zubiaga et al. 2016) dataset containing tweets related to five different events: (i) Charlie Hebdo, (ii) Ferguson, (iii) German wings Crash, (iv) Ottawa Shooting, and (v) Sydney Siege. The dataset includes rumor and non-rumor tweets with reply tweets on those rumor and non-rumor tweets. The overall data statistics can be found in Table 2.

3.2 Model 1: Conventional Machine Learning Models

We extracted thirteen linguistic and user features from tweet to train and test the machine-learning model. The linguistic-based features are (i) question existence in the tweet, (ii) tweet having supportive words, (iii) tweets having denial words, (iv) sentiment of the tweet, (v) length of the tweet, and user characteristic based features are: (vi) verified users or not, (vii) the number of followers, (viii) number of followees, (ix) existence of URL in a tweet, (x) number of hashtags in tweet, (xi) user account registration days, (xii) status count of a tweet, and (xiii) retweet count of a tweet. The complete description of the features is placed in Table 3.

Table 2 Data statistics of rumor and non-rumor classes

News	Rumor	Non-Rumor
Charlie Hebdo	458 (22.0%)	1,621 (78.0%)
Ferguson	284 (24.8%)	859 (75.2)%
Germanwings Crash	238 (50.7%)	231 (49.3%)
Ottawa Shooting	470 (52.8%)	420 (47.2%)
Sydney Siege	522 (42.8%)	699 (57.2%)
Total	1972 (34.0%)	3830 (66%)

The extracted features have different variances that can dominate other features during training the classifiers. So the standardization of data is done for better representation of features for machine learning classifiers. The standardization of features is carried out independently on each of the features in such a way that it has a zero mean and a standard deviation of one. These features are then used by different conventional machine learning classifiers to classify rumor and non-rumor tweets. The detailed results of the different classifiers are shown in Section 4.

3.3 Model 2: Deep Learning Models with PSO

The use of a deep learning model can effectively preserve the contextual information of tweet text and eliminate the requirement of hand-crafted features. The word embedding technique is used to convert each tweet into a fixed vector dimension that is given to deep learning models. The detailed description of the tweet representation can be seen in Section 3.3.1.

3.3.1 Tweet Representation

Word embedding technique is used to represent each tweet in a fixed dimension of a real-valued vector. The word embedding generates a similar vector of the words having similar contextual meanings. The pre-trained GloVe (Pennington et al. 2014) look-up matrix[1] is used to create the embedding vector of 200-dimension for each word W_i. The pre-trained GloVe embedding is used to limit the computation overhead and get better performance as GloVe is trained on the massive corpus of tweets. The complete embedded tweet matrix T_i is represented as:

$$T_i = \begin{array}{c} \begin{matrix} W_1 & W_2 & W_3 & \dots W_m \end{matrix} \\ \begin{bmatrix} e_{11} & e_{21} & e_{31} & \dots & e_{m1} \\ e_{12} & e_{22} & e_{32} & \dots & e_{m2} \\ e_{13} & e_{23} & e_{33} & \dots & e_{m3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ e_{1k} & e_{2k} & e_{3k} & \dots & e_{mk} \end{bmatrix} \end{array}$$

Where T_i is the embedded tweet matrix of a tweet with m words (padding is done if needed). Padding is done to

[1] https://nlp.stanford.edu/data/glove.Twitter.27B.zip

Table 3 List of different features with type and their description

Type	Feature	Description
Linguistic based	Question Existence	Tweet contains question or not (Binary: 0 or 1)
	Tweet having supportive words	Tweet having support words like true, exactly, yes, indeed, omg, and know
	Tweets having denial words	Tweet having denial words like not true, false, impossible, shut, and don't agree
	Length of the tweet	Total number of characters in the tweet text
	Sentiment of the tweet	Sentiment score of the tweets using SentiWordNet dictionary
User based	Verified users or not	Twitter account is verified or not (Binary: 0 or 1)
	Existence of URL in tweet	Tweet contains URL or not (Binary: 0 or 1)
	Number of followees	The number of users who follows an account
	Number of followers	The number of users who was followed by an account
	Number of hashtags in tweet	The number of hashtags in tweet text
	User account registration days	The number of days since user profile was created
	Status count of tweet	The total number of tweets posted by user
	Retweet count of tweet	The number of users repost the tweet

fix the length of each tweet to the same size. The vector $[e_{m1}e_{m2}.....e_{mk}]$ represents the embedding of word W_m, and k represents the embedding dimension. In this work, the value of m is fixed to 32, which means that tweets with more than 32 words are curtailed, and tweets with less than 32 words are padded to make it of 32-word length.

3.3.2 Convolutional Neural Network (CNN)

The CNN models are successfully used in various natural language processing tasks (Kumar and Singh 2019; Chen et al. 2017; Yu et al. 2017). In this work, the convolutional neural network-based model is used to automatically extract the features from the text of a tweet. For our work, a 2-layer of convolution is performed with 128 filters of size 2-gram, 3-gram, and 4-gram. After the 2-layer of CNN, 2-dense layers are used with the 256 and 2 neurons. The detailed model configurations and hyper-parameter settings can be seen in Table 4. The CNN model is trained with tweets for 150 epochs, and the output of the dense layer having 256 neurons are stored as features. This 256-dimensional feature map is concatenated with thirteen linguistic and user features to make it a 269-dimensional hybrid feature set for further processing.

3.3.3 Long-Short Term Memory (LSTM)

The LSTM model with two LSTM layers with 200 and 100-dimensional hidden state vectors for the first and second LSTM layers respectively are used to extract the features from the tweet text. The configuration and hyper-parameter settings for this model can be found in Table 4. The network training is performed with the rumor and non-rumor tweets for 150 epochs. The output of the second layer having a 100-dimensional hidden state vector is stored as features. This 100-dimensional feature map is concatenated with thirteen linguistic and user features to make it a 113-dimensional hybrid feature set for further processing.

3.3.4 Particle Swarm Optimization (PSO)

All features used in a classification task may not be equally effective. Several irrelevant and redundant features can even reduce the performance of machine learning tasks. The feature selection aims to find the small and relevant feature set from the hybrid feature sets to achieve better performance. We used binary PSO (Khanesar et al. 2007) to find the optimal set of features from the hybrid feature set extracted features from CNN and LSTM models with manually extracted 13-features. We used 50 swarm particles and iterated it for 500 iterations. In the case of CNN, PSO reduces 269-dimensional features to 185-dimensional features, and for the LSTM model, it reduces 113-dimensional features to 90-dimensional features. These optimized feature sets are used by different conventional machine learning classifiers listed earlier. The detailed result is placed in Section 4.

3.4 Model 3: Attention-based Long-Short Term Memory (LSTM) Models

Motivated by several successful attention-based techniques (Yang et al. 2016) in natural language processing, we developed a similar model to detect rumor tweets by efficiently learning the distinctive textual features. The attention layer learns the weighting of the input sequence and averages the sequence to obtain the relevant information. The detailed description of the attention-based mechanism can be found in Vaswani et al. (2017). We implemented two different models with LSTM: (i) LSTM + Attention (Text), and (ii)

Table 4 Hyper-parameter settings for each of the deep neural network models

Parameters	CNN	LSTM	LSTM (Attention)
Number of layers	2-CNN, 2-Dense	2-LSTM, 1-Dense	2-LSTM, 1-Dense
Dimension of hidden state vector	–	200, 100	256, 128
Number of filters	128, 128	–	–
Filter size	2, 3,4	–	–
Pooling window	5 (Max)	–	–
Number of neurons (Dense)	256, 2	2	2
Activation	ReLu, Softmax	Softmax	Softmax
Learning rate	0.001	0.001	0.001
Optimizer	Adam	Adam	Adam
Loss	Binary crossentropy	Binary crossentropy	Binary crossentropy
Batch size	4	100	100
Epochs	150	150	150

LSTM + Attention (hybrid features). In the case of the LSTM + Attention (Text) model, the attention layer is used after the second LSTM layer. This model uses only tweet text to train the network to classify tweets for rumors and non-rumors. In the case of LSTM + Attention (hybrid features), after the second layer of the LSTM, attention is used. Then thirteen linguistic and user features are concatenated to the attention layer output. The systematic diagram of the Attention-based LSTM model can be seen in Fig. 1. Finally, the concatenated feature map is used to classify rumor and non-rumor tweets. The detailed hyper-parameter settings for each of the models are shown in Table 4.

4 Results

This section discusses the results of the different models of conventional machine learning, deep learning with PSO, and Attention-based Long-Short Term Memory models. In the case of deep learning with the PSO model, extracted features were divided into training and testing sets with a 3:1 ratio. It means 75% data sample was used for training the classifier, and the remaining 25% data sample was used to test the model. In the case of conventional machine learning and Attention-based LSTM models, 5-fold cross-validation was performed to see the performance of the models. The rationale for using 5-fold cross-validation is that we have a total of 5,804 number of rumor and non-rumor tweets. Therefore with the increase of folds, the testing data samples becomes very small.

4.1 Evaluation Metrics

The performance of the proposed models is evaluated using Precision, Recall, F_1-score, Accuracy and AUC-ROC curve. The description of Precision (P), Recall (R), F_1-score (F_1)

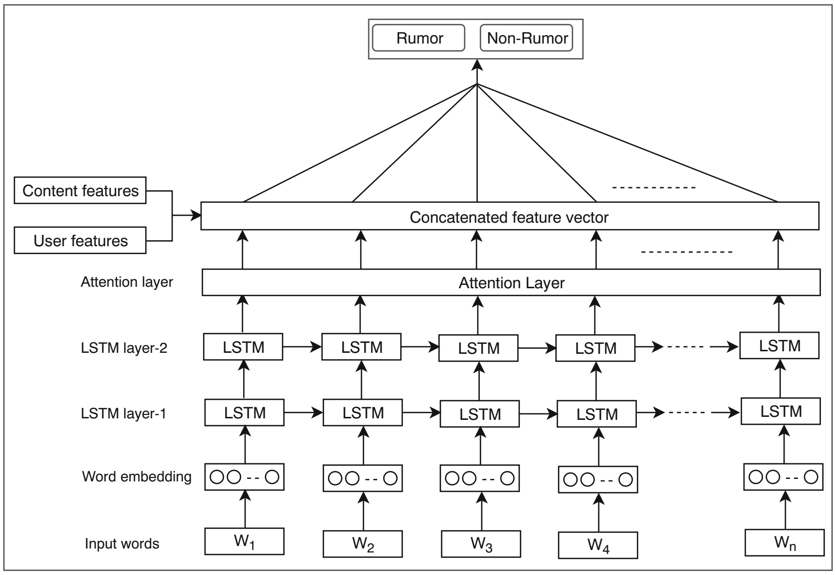

Fig. 1 Proposed Attention based LSTM diagram

and Accuracy (Acc) can be seen from Eqs. 1, 2, 3, 4 respectively. The description of the AUC-ROC curve can be seen from Eqs. 5 and 6. Here, **TP** refers to the number of rumor tweets predicted as rumor, **FP** refers to the number of non-rumor tweets predicted as rumor tweets, **FN** refers to the number of rumor tweet predicted as non-rumor and **TN** refers to the number of non-rumor tweet prected as non-rumor tweet.

- Precision: It is the number of accurately predicted rumor tweets to the total number of predicted rumor tweets.

$$\text{Precision} = \frac{\text{Number of accurately predicted rumor tweets}}{\text{Total number predicted rumor tweets}} = \frac{\text{TP}}{\text{TP + FP}} \tag{1}$$

- Recall: It is the number of accurately predicted rumor tweets to the total number of actual rumor tweets.

$$\text{Recall} = \frac{\text{Number of accurately predicted rumor tweets}}{\text{Total number of actual rumor tweets}} = \frac{\text{TP}}{\text{TP + FN}} \tag{2}$$

- F_1-score: It is the harmonic mean between Precision and Recall. It gives the balanced evaluation between both Precision and Recall.

$$F_1 - \text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision + Recall}} \tag{3}$$

- Accuracy: Accuracy is defined as the ratio of TP + TN to the total data set.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+FP+FN+TN}} \tag{4}$$

- AUC-ROC curve: It is knows as Area Under The Curve - Receiver Operating Characteristics. ROC curve is plotted with True Positive Rate (TPR) against the False Positive Rate (FPR). True Positive Rate (TPR) and False Positive Rate (FPR) can be defined as:

$$\text{True Postive Rate (TPR)} = \frac{\text{TP}}{\text{TP + FN}} \tag{5}$$

$$\text{False Positive Rate (FPR)} = \frac{\text{FP}}{\text{TN + FP}} \tag{6}$$

4.2 Results of Model 1: Conventional Machine Learning Models

We started the experiments with conventional machine learning classifiers using thirteen linguistic and user features extracted from the tweets, explained in Table 3. The performance of seven different conventional machine learning classifiers (i) Support Vector Machine (SVM), (ii) Random Forest (RF), (iii) Logistic Regression (LR), (iv) K-Nearest Neighbor (KNN), (v) Naive Bayes (NB), (vi) Gradient Boosting (GB), and (vii) Decision Tree (DT) with 5-fold cross-validation is shown in Table 5.

Out of the seven machine learning classifiers, the Logistic Regression classifier performed worst for the rumor (R) class. It achieved a recall of 0.13 and a F_1-score of 0.21, whereas, the Random Forest classifier performed best with the accuracy of 73% as shown in bold in Table 5. The Random Forest classifier achieved a recall of 0.88 and 0.42 for non-rumor and rumor class, respectively, as shown in bold in Table 5. It means for the rumor class, out of total 100 rumor tweets, Random Forest classifier was only able to classify 42 tweets as the rumor. As our target is to identify rumor tweets, the performance of these conventional machine learning classifiers with thirteen linguistic and user features was quite low. Therefore, we moved to deep learning-based models as the deep learning models could automatically learn better features to distinguish rumor and non-rumor tweets.

4.3 Results of Model 2: Deep Learning Models with PSO

We extracted 256 and 100-dimensional feature vectors from tweet text using CNN and LSTM models respectively. PSO was applied to the extracted features of deep learning models and thirteen linguistic and user features to extract the more relevant feature subset as explained in Section 3. For features obtained from the CNN model, we got a 185-dimensional optimized feature set, whereas, for the LSTM model, we got a 90-dimensional optimized feature set. This optimized feature set is then used with the conventional machine learning classifiers SVM, RF, LR, KNN, NB, GB, and DT to classify rumor and non-rumor tweets. The results for the different classifiers used on the optimized feature set can be seen from Table 6.

Support Vector Machine (SVM) classifier applied on features extracted through CNN + PSO model performed worst and achieved only a recall of 0.46 for rumor class. Naive Bayes classifier performed best for the rumor class and achieved a recall of 0.79, and overall it achieved an accuracy of 83%. For both the rumor and non-rumor classes, the KNN classifier performed best with the accuracy of 84%, but recall of rumor class is degraded by 4% in the comparison to the Naive Bayes classifier. The automatically extracted feature from CNN + PSO with conventional machine learning classifiers performed better than the classifiers trained with thirteen manually extracted features, as can be seen from Tables 5 and 6.

Similarly, we used all the conventional machine learning classifiers with the automatically extracted features from LSTM + PSO model. All the conventional machine learning classifiers performed quite well in the comparison of the features extracted from the CNN + PSO model. The best

Table 5 Performance of conventional machine learning classifiers with 5-fold cross-validation using linguistic and user features

Classifier	Class	Precision	Recall	F_1-score	Accuracy
SVM	NR	0.69	0.93	0.79	0.68
	R	0.60	0.20	0.30	
Random Forest	NR	**0.75**	**0.88**	**0.81**	**0.73**
	R	**0.65**	**0.42**	**0.51**	
Logistic Regression	NR	0.68	0.93	0.78	0.66
	R	0.50	0.13	0.21	
KNN	NR	0.71	0.80	0.75	0.65
	R	0.49	0.39	0.43	
Naive Bayes	NR	0.74	0.59	0.64	0.59
	R	0.43	0.58	0.48	
Gradient Boosting	NR	0.72	0.72	0.72	0.64
	R	0.47	0.47	0.47	
Decision Tree	NR	0.72	0.71	0.72	0.63
	R	0.46	0.47	0.46	

performance we got in case of Decision Tree classifier where we achieved a recall of 0.81 and F_1-score of 0.81 for rumor class and overall it achieved an accuracy of 86%, as can be seen from Table 6.

4.4 Results of Model 3: Attention based LSTM Models

Next, we applied attention technique with LSTM model to build two types of different models: (i) only tweet texts were used for LSTM along with attention denoted as LSTM + Attention (Text) (ii) tweet texts with thirteen linguistic and user features for LSTM along with attention denoted as LSTM + Attention (hybrid features). We performed 5-fold cross validation for both the models. The class-wise results for rumor and non-rumor class for both the deep learning based models can be seen from Table 6. The box-whisker plot shown in Fig. 2 showed that the F_1-score for 5-fold cross validation of both the LSTM + Attention (Text) and LSTM + Attention (hybrid features) models are in the range of 0.85-0.87 and 0.86-0.89 respectively. As can be seen from the Table 6 and Fig. 2, the attention based LSTM model with hybrid features outperformed all the existing models. The attention based LSTM with hybrid features achieved aprecision of 0.82, recall of 0.81, and F_1-score of 0.82 for rumor class and precision, recall and F_1-score of 0.90, 0.91, and 0.91 respectively for non-rumor class as shown in bold in Table 6. It can be seen from Fig. 2, the F_1-score varies from 0.86 to 0.89 when it was validated using 5-fold cross validation and achieved an average F_1-score of 0.88. We plotted fold wise confusion matrix and AUC-ROC curve for the best performed LSTM + Attention (hybrid features). The fold wise confusion matrix can be seen from Figs. 3 and 4. Figures 5, 6, and 7 for fold-1, fold-2, fold-3, fold-4, fold-5 respectively. The fold wise AUC-ROC curve can be seen from Figs. 8, 9, 10, 11, and 12 for fold-1, fold-2, fold-3, fold-4, and fold-5 respectively. In AUC-ROC curve, class 0 represent non-rumor class and class 1 represent rumor class.

5 Discussion

The major finding of the current research is that an Attention-based LSTM model is performing better than all the existing models to classify tweets into rumor or non-rumor class. Another finding is that a hybrid model using PSO based feature optimization also yields similar results for the rumor class but inferior results than the Attention-based LSTM model for the non-rumor class. The reason was that the fitness of the PSO algorithm was optimized for extracting the most relevant features from rumorous tweets only. The manually extracted thirteen linguistic and user features played a significant role in identifying the rumourous tweet as shown in Fig. 2. Without those features, the model reported average F1-score of around 86% whereas with these features the average F1-score was increased to 88%. The deep learning models are found to perform better compared to machine learning classifiers. The best result of machine learning classifiers was having an accuracy of 73% whereas the deep learning model was having an accuracy of 88%. Among the deep learning models, LSTM was found to perform better than CNN models as LSTM was able to capture the sequence information better than CNN models. The Attention mechanism was found to be very effective as it was performing better than any other model which can be seen from Table 7. The performance of all the models was better for the non-rumor class compared to the rumor class due to the higher number of samples in the non-rumor class. The rumor class has only 34% of data samples, whereas the non-rumor class has 66% data samples.

Table 6 Class-wise results for the implemented models

Models		Class	Precision	Recall	F_1-score	Accuracy
CNN + PSO (Optimized no. of features = 185)	SVM	NR	0.75	0.92	0.83	0.75
		R	0.76	0.46	0.57	
	RF	NR	0.86	0.90	0.88	0.84
		R	0.81	0.73	0.77	
	LR	NR	0.86	0.89	0.88	0.84
		R	0.79	0.74	0.77	
	KNN	NR	0.86	0.89	0.88	0.84
		R	0.79	0.75	0.77	
	NB	NR	0.88	0.85	0.86	0.83
		R	0.75	0.79	0.77	
	GB	NR	0.85	0.90	0.88	0.84
		R	0.81	0.73	0.77	
	DT	NR	0.85	0.90	0.88	0.84
		R	0.81	0.72	0.76	
LSTM + PSO (Optimized no. of features = 90)	SVM	NR	0.88	0.90	0.89	0.86
		R	0.81	0.79	0.80	
	RF	NR	0.89	0.89	0.89	0.86
		R	0.80	0.80	0.80	
	LR	NR	0.89	0.89	0.89	0.86
		R	0.81	0.80	0.80	
	KNN	NR	0.88	0.90	0.89	0.86
		R	0.82	0.79	0.80	
	NB	NR	0.89	0.89	0.89	0.86
		R	0.81	0.80	0.81	
	GB	NR	0.89	0.88	0.89	0.86
		R	0.80	0.81	0.80	
	DT	NR	0.89	0.89	0.89	0.86
		R	0.80	0.81	0.81	
LSTM + Attention (Text)		NR	0.90	0.89	0.89	0.86
		R	0.80	0.79	0.79	
LSTM + Attention (hybrid feature)		NR	**0.90**	**0.91**	**0.91**	**0.88**
		R	**0.82**	**0.81**	**0.82**	

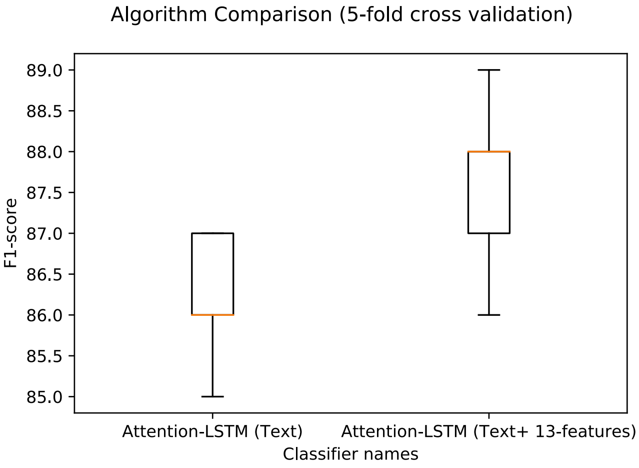

Fig. 2 Deep learning classifier comparison for the 5-fold cross validation

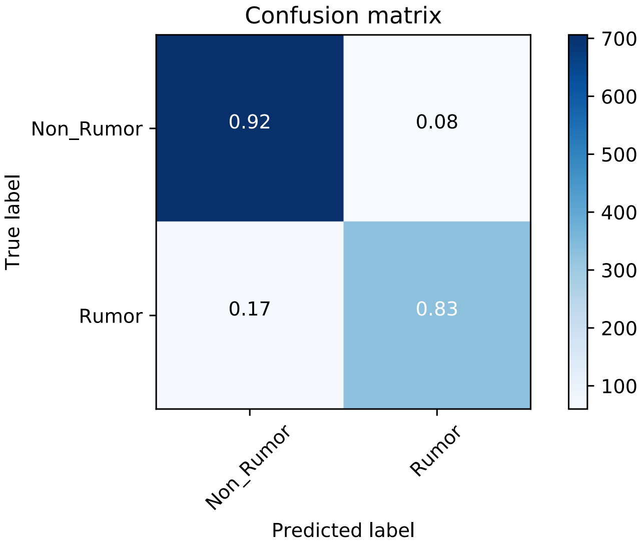

Fig. 3 Confusion matrix for fold-1 in case LSTM + Attention (Text + features)

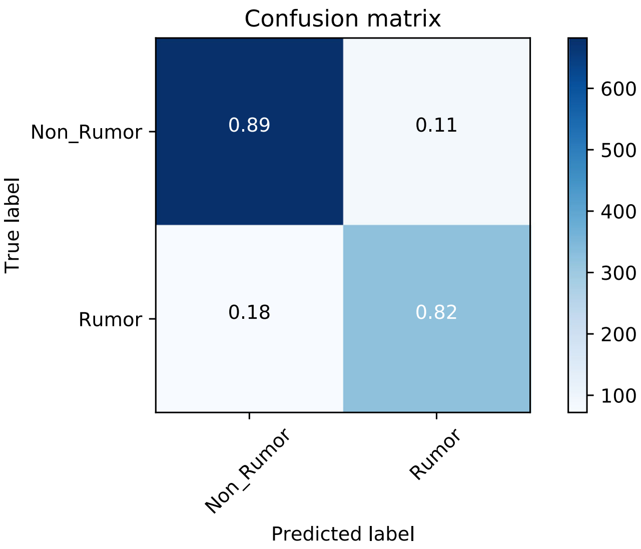

Fig. 4 Confusion matrix for fold-2 in case LSTM + Attention (Text + features)

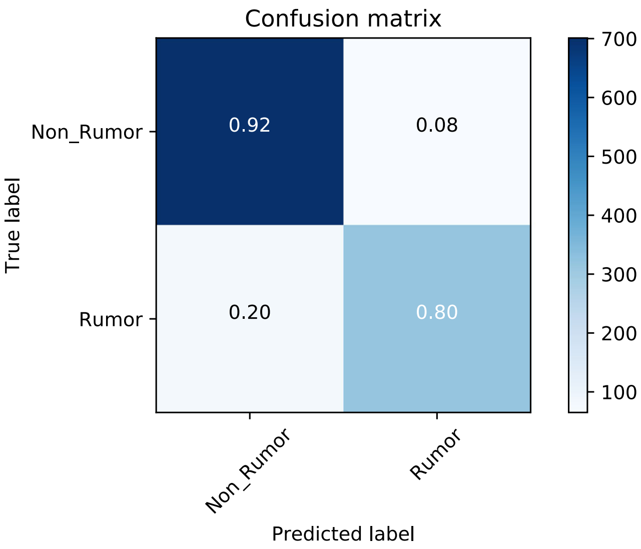

Fig. 6 Confusion matrix for fold-4 in case LSTM + Attention (Text + features)

The findings of our research are in line with similar works by Zubiaga et al. (2016), Ma et al. (2016), Yu et al. (2017), Ajao et al. (2018), and Asghar et al. (2019). A comparative result of our models with earlier works implemented on the same Pheme dataset (Zubiaga et al. 2016) is shown in Table 7. The first result on the same dataset was reported by Zubiaga et al. (2016) having a precision of 0.67, recall of 0.56, and an F_1-score of 0.61 using Condition Random Field (CRF) with manually extracted linguistic and user features. A Recurrent Neural Network (RNN) based model developed by Ma et al. (2016) achieved a precision of 0.81, recall of 0.81, and F_1-score of 0.80. The CNN based model developed by Yu et al. (2017) achieved a precision of 0.80, recall of 0.80, and F_1-score of 0.78. A precision of 0.83, recall of 0.84, and F_1-score of 0.83 was reported by Ajao et al. (2018) with LSTM-CNN based hybrid model. Asghar et al. (2019) reported precision, recall, and F_1-score of 0.86 using bidirectional LSTM-CNN model. In line with this study, our proposed attention-based LSTM model using hybrid features achieves a precision, recall, and F_1-score of 0.88 as shown in bold in Table 7 which is better than the existing state-of-the-art results.

Our findings are also consistent with the finding of Kim et al. (2019) and Kim and Dennis (2019). Kim and Dennis (2019) find a strong positive correlation with the user's pre-existing beliefs and believability about an article. In our analysis, it is found that when the features such as count of supportive or denial words which aligns a tweet with user's

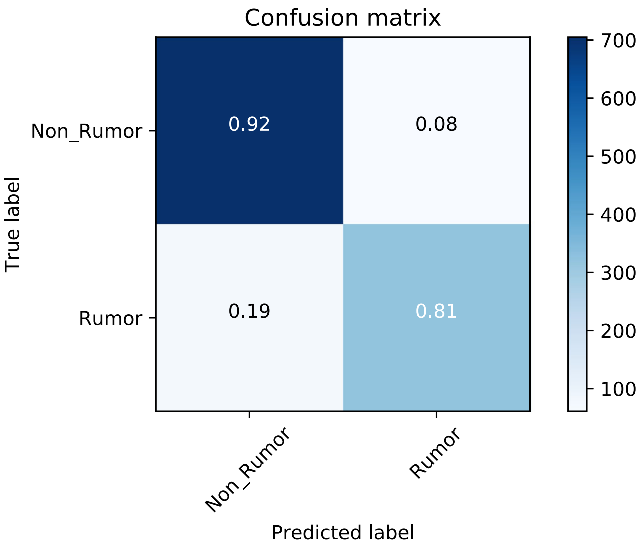

Fig. 5 Confusion matrix for fold-3 in case LSTM + Attention (Text + features)

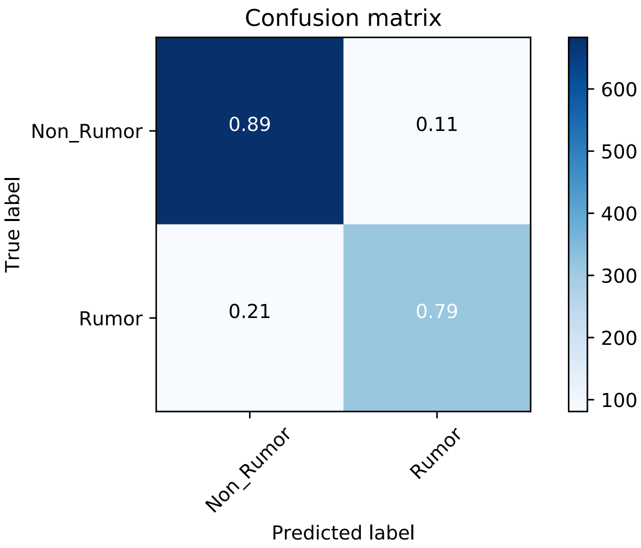

Fig. 7 Confusion matrix for fold-5 in case LSTM + Attention (Text + features)

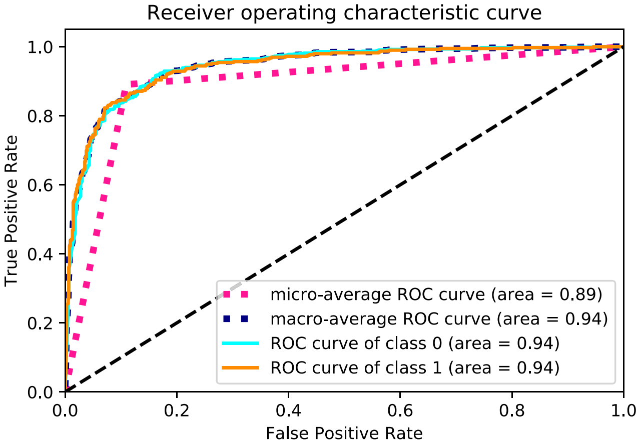

Fig. 8 ROC for fold-1 in case LSTM + Attention (Text + features)

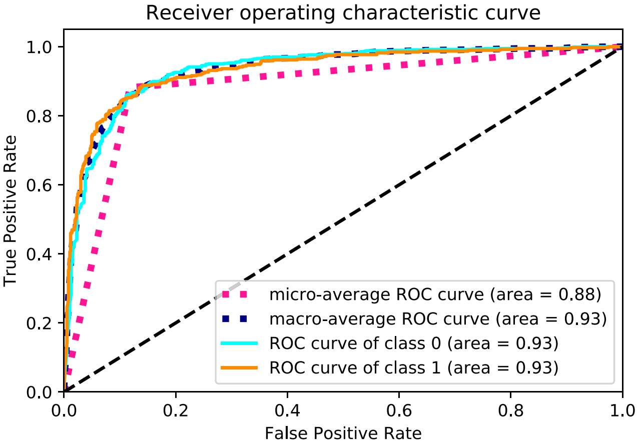

Fig. 10 ROC for fold-3 in case LSTM + Attention (Text + features)

pre-existing beliefs increases the classification accuracy. The user account related features such as account verified or not (Vosoughi et al. 2017), age of account, number of tweets through the account improves the classification accuracy which is very similar to the source rating proposed by Kim et al. (2019). Kim et al. (2019) and Kim and Dennis (2019) used Facebook and news sources for their study where source rating was available but we used Twitter where source rating can be approximated by age of account, number of tweets through the account, number of followers and followee and retweet count of tweets. A detailed description of the features are presented in Table 3.

5.1 Theoretical Contributions

The major theoretical contributions of the present research are the attention-based LSTM model for rumorous tweet identification. The attention mechanism was able to capture the text highlighting rumor behavior. The attention layer is a sequential neural network layer that focuses on the specific words of the input which are present in a specific class tweet. In this article, we have a simple additive attention layer to capture the relevant words of a specific class.

The other contribution was making a hybrid feature set by extracting linguistic and user features from tweets manually and text features through deep learning models. The linguistic and user features alone have been used by several machine learning classifiers to achieve very limited success as shown in Table 5. On the other hand, deep learning models with automatic feature extracted from the text was also found to reach a limit as can be seen in Table 7. The hybrid deep learning models with a combination of LSTM (BiLSTM) and CNN by Ajao et al. (2018) and Asghar et al. (2019) achieved a precision of 0.83 and 0.86 respectively. Even our attention model also achieved the same result with a precision of 0.86. But the hybridization of the feature improves the results to 0.88 in terms of precision. Hence, it confirms that hybridization works well for the said task.

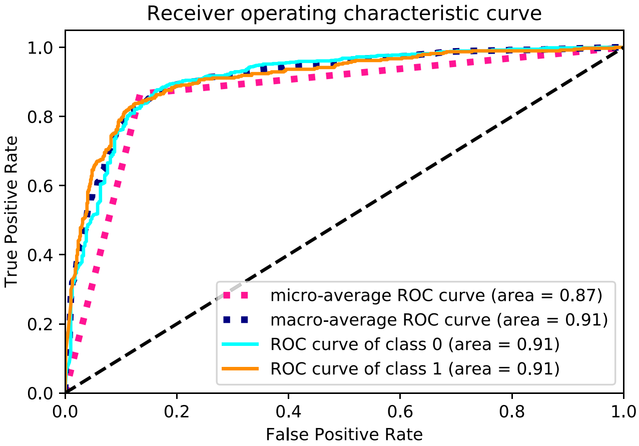

Fig. 9 ROC for fold-2 in case LSTM + Attention (Text + features)

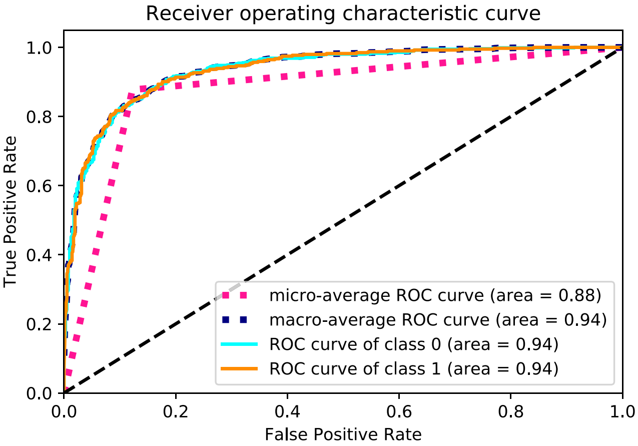

Fig. 11 ROC for fold-4 in case LSTM + Attention (Text + features)

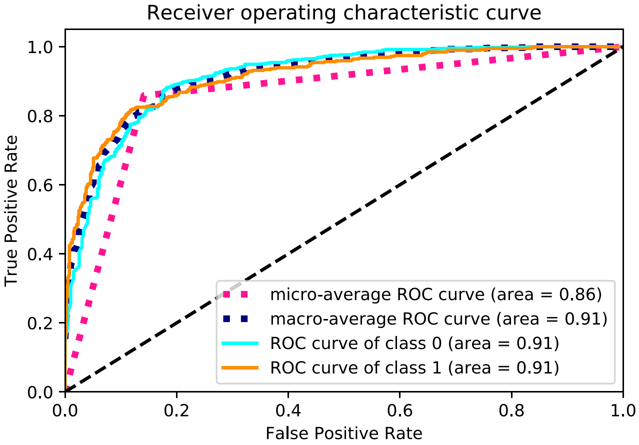

Fig. 12 ROC for fold-5 in case LSTM + Attention (Text + features)

One more theoretical contribution was the model for the optimization of the hybrid feature space through the PSO algorithm. The model did improve the results of predicting the rumor class by any other existing models. Although, the model reports lesser value in terms of accuracy compared to the LSTM with the Attention model using hybrid features. The results confirm that the PSO optimization worked well for the target rumor class.

5.2 Implications for Practice

The proposed Attention-based LSTM model can identify rumorous tweets as early as possible with significant accuracy, which can help to debunk the spread of rumor at a very early stage. This proposed system can reduce the impact of rumors on society and weaken the loss of life, money, and build the firm trust of users with social media platforms. One of the practical implications of the proposed system is that it can be developed as an application for a smartphone that can classify posted tweets into rumor and non-rumor classes. While retweeting a rumor, a message may be given to the user that the tweet may be a rumor. It can reduce the propagation of rumors on Twitter.

The main limitations of the current system are that only textual information of a tweet along with some user features are considered for this research. The other components of a tweet such as images, audio, video, animated Graphics Interchange Formats (GIFs), memes and URLs may also help to identify rumor tweets. The other limitation of the current research is that it is only validated with English tweets. With other languages and multi-lingual tweets that are very common in several non-English speaking countries, the model may not produce similar results. The current system may be extended to align properly with the guidelines of design science research (March and Smith 1995; Baskerville et al. 2018) in the future.

5.3 Limitations and Future Research

One of the limitations of the current research is that we have only validated the model with a dataset from one social media site. This may limit the applicability of the current model with other social media sites. Second, we only focused on the textual content and some user features. Even though text is is the most commonly used medium to propagate rumors, there are other components of a rumor such as images, video, and emoticons, etc. For a complete rumor detection system, images, videos, emoticons, etc. should also be included as features. The other limitation of the proposed work is that it is language-dependent as the system is trained and validated with English language tweets only. It may not perform equally well with tweets containing bi-lingual or multilingual comments.

In future research, the dataset from different social media may be collected to properly validate the results to generalize the result of the model. In the future, the Uniform Resource Locator (URL), emoticons, images, and videos along with the text may be used as features too. The proposed model is a supervised model which requires a lot of labeled dataset for proper training and validation. In the future, unsupervised models and Generative Adversarial Networks may be developed to eliminate or reduce the need for a labeled dataset.

Table 7 Comparison of the proposed work with the existing works

Authors	Approach	Feature	P	R	F_1	Acc
Zubiaga et al. (2016)	CRF classifier	Content + Social	0.67	0.56	0.61	-
Ma et al. (2016)	RNN	Word embedding	0.81	0.81	0.80	80.86
Yu et al. (2017)	CNN	Word embedding	0.80	0.80	0.78	79.74
Ajao et al. (2018)	LSTM-CNN	Word embedding	0.83	0.84	0.83	83.53
Asghar et al. (2019)	BiLSTM-CNN	Word embedding	0.86	0.86	0.86	86.12
Proposed	LSTM+PSO	Hybrid Features	**0.86**	**0.86**	**0.86**	**86.00**
Proposed	LSTM+Attention	Hybrid Features	**0.88**	**0.88**	**0.88**	**88.00**

6 Conclusion

Rumor veracity estimation from the tweet is a critical task on Twitter. In this study, we have implemented and compared the performance of several machine and deep learning-based models to identify rumorous tweets at a very early stage. A hybrid feature set is created by extracting thirteen linguistic and user features from the tweets and one hundred features were extracted from text using the LSTM model. Machine learning models were trained with thirteen linguistic and user feature whereas the deep learning models were trained with hybrid features. A population-based optimization algorithm was employed to select the optimal number of features from the hybrid feature set which was able to reduce the total features by more than 20%. The experimental results proved the effectiveness of the deep learning-based model over conventional machine learning models for rumor identification. The proposed LSTM with Attention model with hybrid features outperformed all the existing models with the F_1-score of 0.88.

References

Abedin, B., & Babar, A. (2018). Institutional vs. non-institutional use of social media during emergency response: A case of Twitter in 2014 Australian bush fire. *Information Systems Frontiers*, *20*, 729–740.

Ajao, O., Bhowmik, D., Zargari, S. (2018). Fake news identification on Twitter with hybrid cnn and rnn models. In *Proceedings of the 9th International Conference on Social Media and Society* (pp. 226–230): ACM.

Alalwan, A.A., Rana, N.P., Dwivedi, Y.K., Algharabat, R. (2017). Social media in marketing: a review and analysis of the existing literature. *Telematics and Informatics*, *34*, 1177–1190.

Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, *31*, 211–36.

Alryalat, M.A.A., Rana, N.P., Sahu, G.P., Dwivedi, Y.K., Tajvidi, M. (2017). Use of social media in citizen-centric electronic government services: a literature analysis. *International Journal of Electronic Government Research (IJEGR)*, *13*, 55–79.

Asghar, M.Z., Habib, A., Habib, A., Khan, A., Ali, R., Khattak, A. (2019). Exploring deep neural networks for rumor detection. *Journal of Ambient Intelligence and Humanized Computing*, 1–19.

Aswani, R., Kar, A.K., Ilavarasan, P.V. (2018). Detection of spammers in Twitter marketing: a hybrid approach using social media analytics and bio inspired computing. *Information Systems Frontiers*, *20*, 515–530.

Baabdullah, A.M., Rana, N.P., Alalwan, A.A., Algharabat, R., Kizgin, H., Al-Weshah, G.A. (2018). Toward a conceptual model for examining the role of social media on social customer relationship management (SCRM) system. In *International Working Conference on Transfer and Diffusion of IT* (pp. 102–109): Springer.

Baskerville, R., Baiyere, A., Gregor, S., Hevner, A., Rossi, M. (2018). Design science research contributions: Finding a balance between artifact and theory. *Journal of the Association for Information Systems*, *19*, 3.

Castillo, C., Mendoza, M., Poblete, B. (2011). Information credibility on Twitter. In *Proceedings of the 20th international conference on World Wide Web* (pp. 675–684): ACM.

Chen, T., Li, X., Yin, H., Zhang, J. (2018a). Call attention to rumors: Deep attention based recurrent neural networks for early rumor detection. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 40–52): Springer.

Chen, W., Zhang, Y., Yeo, C.K., Lau, C.T., Lee, B.S. (2018b). Unsupervised rumor detection based on users' behaviors using neural networks. *Pattern Recognition Letters*, *105*, 226–233.

Chen, Y.-C., Liu, Z.-Y., Kao, H.-Y. (2017). Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 465–469).

Derczynski, L., Bontcheva, K., Liakata, M., Procter, R., Hoi, G.W.S., Zubiaga, A. (2017). Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 69–76).

DiFonzo, N., & Bordia, P. (2007). Defining rumor. *Rumor psychology : Social and organizational approaches*, 11–34. https://doi.org/doi.org/10.1037/11503-001.

Dwivedi, Y.K., Kapoor, K.K., Chen, H. (2015). Social media marketing and advertising. *The Marketing Review*, *15*, 289–309.

Enayet, O., & El-Beltagy, S.R. (2017). Niletmrg at semeval-2017 task 8: Determining rumour and veracity support for rumours on Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 470–474).

Ghosh, S., Ghosh, K., Ganguly, D., Chakraborty, T., Jones, G.J., Moens, M.-F., Imran, M. (2018). Exploitation of social media for emergency relief and preparedness: Recent research and trends. *Information Systems Frontiers*, *20*, 901–907.

Hamidian, S., & Diab, M. (2016). Rumor identification and belief investigation on Twitter. In *Proceedings of the 7th Workshop on computational approaches to subjectivity, sentiment and social media analysis* (pp. 3–8).

Huang, H. (2017). A war of (mis) information: The political effects of rumors and rumor rebuttals in an authoritarian country. *British Journal of Political Science*, *47*, 283–311.

Jain, A., Borkar, V., Garg, D. (2016). Fast rumor source identification via random walks. *Social Network Analysis and Mining*, *6*, 62.

Kapoor, K.K., Tamilmani, K., Rana, N.P., Patil, P., Dwivedi, Y.K., Nerur, S. (2018). Advances in social media research: past, present and future. *Information Systems Frontiers*, *20*, 531–558.

Khan, M.L., & Idris, I.K. (2019). Recognise misinformation and verify before sharing: a reasoned action and information literacy perspective. *Behaviour & Information Technology*, *38*, 1194–1212.

Khanesar, M.A., Teshnehlab, M., Shoorehdeli, M.A. (2007). A novel binary particle swarm optimization. In *2007 Mediterranean Conference on Control & Automation* (pp. 1–6): IEEE.

Kim, A., & Dennis, A.R. (2019). Says who? the effects of presentation format and source rating on fake news in social media. *MIS Quarterly*, 43.

Kim, A., Moravec, P.L., Dennis, A.R. (2019). Combating fake news on social media with source ratings: The effects of user and expert reputation ratings. *Journal of Management Information Systems*, *36*, 931–968.

Kizgin, H., Jamal, A., Dey, B.L., Rana, N.P. (2018). The impact of social media on consumers' acculturation and purchase intentions. *Information Systems Frontiers*, *20*, 503–514.

Kumar, A., & Rathore, N.C. (2016). Relationship strength based access control in online social networks. In *Proceedings of first international conference on information and communication technology for intelligent systems: Volume 2* (pp. 197–206): Springer.

Kumar, A., & Singh, J.P. (2019). Location reference identification from tweets during emergencies: a deep learning approach. *International Journal of Disaster Risk Reduction*, *33*, 365–375.

Kumar, A., Singh, J.P., Dwivedi, Y.K., Rana, N.P. (2020). A deep multi-modal neural network for informative Twitter content classification during emergencies. *Annals of Operations Research*, 1–32. https://doi.org/10.1007/s10479-020-03514-x.

Kumar, A., Singh, J.P., Rana, N.P. (2017). Authenticity of geo-location and place name in tweets. In *Proceedings of 23rd Americas Conference on Information Systems (AMCIS)*.

Kwon, S., Cha, M., Jung, K. (2017). Rumor detection over varying time windows. *PloS one*, *12*, e0168344.

Lee, J., Agrawal, M., Rao, H.R. (2015). Message diffusion through social network service: The case of rumor and non-rumor related tweets during boston bombing 2013. *Information Systems Frontiers*, *17*, 997–1005.

Liang, G., He, W., Xu, C., Chen, L., Zeng, J. (2015). Rumor identification in microblogging systems based on users' behavior. *IEEE Transactions on Computational Social Systems*, *2*, 99–108.

Liu, Y., Jin, X., Shen, H. (2019). Towards early identification of online rumors based on long short-term memory networks. *Information Processing & Management*, *56*, 1457–1467.

Liu, Y., Jin, X., Shen, H., Cheng, X. (2017). Do rumors diffuse differently from non-rumors? a systematically empirical analysis in sina weibo for rumor identification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (pp. 407–420): Springer.

Lozano, M.G., Brynielsson, J., Franke, U., Rosell, M., Tjörnhammar, E., Varga, S., Vlassov, V. (2020). Veracity assessment of online data. *Decision Support Systems*, *129*, 113–132.

Lukasik, M., Srijith, P., Vu, D., Bontcheva, K., Zubiaga, A., Cohn, T. (2016). Hawkes processes for continuous time sequence classification: an application to rumour stance classification in Twitter. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (Vol. 2 pp. 393–398).

Ma, J., Gao, W., Mitra, P., Kwon, S., Jansen, B.J., Wong, K.-F., Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks. In *Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI)* (pp. 3818–3824).

Ma, J., Gao, W., Wei, Z., Lu, Y., Wong, K.-F. (2015). Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (pp. 1751–1754): ACM.

Ma, J., Gao, W., Wong, K.-F. (2017). Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (Vol. 1 pp. 708–717).

March, S.T., & Smith, G.F. (1995). Design and natural science research on information technology. *Decision support systems*, *15*, 251–266.

Meel, P., & Vishwakarma, D.K. (2019). 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*, 112986.

Mondal, T., Pramanik, P., Bhattacharya, I., Boral, N., Ghosh, S. (2018). Analysis and early detection of rumors in a post disaster scenario. *Information Systems Frontiers*, *20*, 961–979.

Oh, O., Agrawal, M., Rao, H.R. (2011). Information control and terrorism: Tracking the mumbai terrorist attack through Twitter. *Information Systems Frontiers*, *13*, 33–43.

Oh, O., Gupta, P., Agrawal, M., Rao, H.R. (2018). ICT Mediated rumor beliefs and resulting user actions during a community crisis. *Government Information Quarterly*, *35*, 243–258.

Pennington, J., Socher, R., Manning, C. (2014). Glove: Global vectors for word representation. In *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543).

Qazvinian, V., Rosengren, E., Radev, D.R., Mei, Q. (2011). Rumor has it: Identifying misinformation in microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1589–1599). Association for Computational Linguistics.

Rath, B., Gao, W., Ma, J., Srivastava, J. (2017). From retweet to believability: Utilizing trust to identify rumor spreaders on Twitter. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017* (pp. 179–186): ACM.

Sammut, C., & Webb, G.I. (2010). Tf–idf. In *Encyclopedia of machine learning* (pp. 986–987). Boston: Springer.

Serrano, E., Iglesias, C.A., Garijo, M. (2015). A survey of Twitter rumor spreading simulations. In *Computational Collective Intelligence* (pp. 113–122): Springer.

Shareef, M.A., Mukerji, B., Dwivedi, Y.K., Rana, N.P., Islam, R. (2019). Social media marketing: Comparative effect of advertisement sources. *Journal of Retailing and Consumer Services*, *46*, 58–69.

Singh, J.P., Dwivedi, Y.K., Rana, N.P., Kumar, A., Kapoor, K.K. (2019a). Event classification and location prediction from tweets during disasters. *Annals of Operations Research*, *283*, 737–757.

Singh, J.P., Rana, N.P., Dwivedi, Y.K. (2019b). Rumour veracity estimation with deep learning for Twitter. In Dwivedi, Y., Ayaburi, E., Boateng, R., Effah, J. (Eds.) *ICT Unbounded, Social Impact of Bright ICT Adoption* (pp. 351–363). Cham: Springer International Publishing.

Singh, P., Dwivedi, Y.K., Kahlon, K.S., Sawhney, R.S., Alalwan, A.A., Rana, N.P. (2019c). Smart monitoring and controlling of government policies using social media and cloud computing. *Information Systems Frontiers*, 1–23.

Smith, K.S., McCreadie, R., Macdonald, C., Ounis, I. (2018). Regional sentiment bias in social media reporting during crises. *Information Systems Frontiers*, *20*, 1013–1025.

Srivastava, A., Rehm, G., Schneider, J.M. (2017). Dfki-dkt at semeval-2017 task 8: rumour detection and classification using cascading heuristics. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)* (pp. 486–490).

Tamilmani, K., Rana, N., Alryalat, M., Alkuwaiter, W., Dwivedi, Y. (2018). Social media research in the context of emerging markets: an analysis of literature published in senior scholars' basket of is journals. *Journal of Advances in Management Research*, *15*, 115–129.

Vallejos, S., Alonso, D.G., Caimmi, B., Berdun, L., Armentano, M.G., Soria, Á. (2020). Mining social networks to detect traffic incidents. *Information Systems Frontiers*, 1–20.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.) *Advances in Neural Information Processing Systems 30* (pp. 5998–6008): Curran Associates Inc.

Vosoughi, S., Mohsenvand, M.N., Roy, D. (2017). Rumor gauge: PredICTing the veracity of rumors on Twitter. *ACM transactions on knowledge discovery from data (TKDD)*, *11*, 1–36.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1480–1489).

Yu, F., Liu, Q., Wu, S., Wang, L., Tan, T. (2017). A convolutional approach for misinformation identification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence IJCAI'17* (pp. 3901–3907): AAAI Press.

Zhao, Z., Resnick, P., Mei, Q. (2015). Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 1395–1405). International World Wide Web Conferences Steering Committee.

Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., Procter, R. (2018). Detection and resolution of rumours in social media: a survey. *ACM Computing Surveys (CSUR)*, *51*, 32.

Zubiaga, A., Liakata, M., Procter, R., Hoi, G.W.S., Tolmie, P. (2016). Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, *11*, e0150989.

Jyoti Prakash Singh is an Assistant Professor in the Department of Computer Science and Engineering in National Institute of Technology Patna, India. He has co-authored seven textbooks in the area of C programming, Data Structures, Operating systems and Ad Hoc Networks. Apart from this. He has more than 30 international journal publications and 50 international conference proceedings. His research interests focus on making social media mining, deep learning, social network and information security. He is a senior member of IEEE and ACM, Life member of Computer Society of India (CSI) and Indian Society of Technical Society (ISTE) and Institution of Engineers (IE). He is an Associate Editor of International Journal of Electronic Government Research.

Abhinav Kumar obtained a BTech in Computer Science and Engineering from Gaya College of Engineering in 2013. He attained an MTech in Computer Science from Central University of South Bihar in 2015. He is currently pursuing his PhD in the Department of Computer Science and Engineering of National Institute of Technology Patna, India. He has nine research publications in journals and international conferences. His research interest includes crisis informatics, geoscience, text mining, deep learning, and social networks. He is a student member of IEEE.

Nripendra P. Rana is a Professor in Digital Marketing and the Head of International Business, Marketing and Branding at the School of Management at University of Bradford, UK. His current research interests focus primarily on adoption and diffusion of emerging ICTs, e-commerce, m-commerce, e-government and digital and social media marketing. He has published more than 200 papers in a range of leading academic journals, conference proceedings, books etc. He has co-edited five books on digital and social media marketing, emerging markets and supply and operations management. He has also co-edited special issues, organised tracks, mini-tracks and panels in leading conferences. He is a Chief Editor of International Journal of Electronic Government Research and Associate Editor of International Journal of Information Management. He is a Senior Fellow of the Higher Education Academy (SFHEA) in the UK. He is also a Visiting Scholar at Indian Institute of Management Tiruchirappalli in India.

Yogesh K. Dwivedi is a Professor of Digital Marketing and Innovation, Founding Director of the Emerging Markets Research Centre (EMaRC) and Co-Director of Research at the School of Management, Swansea University, Wales, UK. Professor Dwivedi is also currently leading the International Journal of Information Management as its Editor-in-Chief. His research interests are at the interface of Information Systems (IS) and Marketing, focusing on issues related to consumer adoption and diffusion of emerging digital innovations, digital government, and digital and social media marketing particularly in the context of emerging markets. Professor Dwivedi has published more than 300 articles in a range of leading academic journals and conferences that are widely cited (more than 18 thousand times as per Google Scholar). Professor Dwivedi is an Associate Editor of the Journal of Business Research, European Journal of Marketing, Government Information Quarterly and International Journal of Electronic Government Research, and Senior Editor of the Journal of Electronic Commerce Research. More information about Professor Dwivedi can be found at: http://www.swansea.ac.uk/staff/som/academic-staff/y.k.dwivedi.