IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

RUMOR DETECTION ON TWITTER USING GRAPH CONVOUTIONAL NETWORK

[1]S.Ramapriya, [2]J.Sudha

[1]PG Student, [2]Associate Professor
[1]Computer Science and Engineering,
[1]A.V.C College of Engineering , Mayiladuthurai, India

ABSTRACT: The authenticity of information has become a long standing issue affecting businesses and society in social media. On social networks, the reach and effects of information spread occur at such a fast rate and spreading false information on social media may cause real world impacts. Rumors on social media may cause public panic and negative impact on individuals. So it is necessary to makes the automatic detection of rumors and blocking the user who continuously spreads rumor. This system detect rumor by learning user representation(User behaviour) by graph convolutional networks(GCN) and learning content semantics and propagation clues of rumored tweets. GCN efficiently capture node features and graph structure features in graph structure data and it helps to learn the user information. The proposed system uses user based features, content based features, propagation based features to detect rumors. The main objectives of this system is develop a social network without any fake news spreading. It taken twitter based social network to predict the fake news spreading system. In this system fake news detection is done by using Recurrent Neura Network and Graph Convolutional Network. This model experiments on two real-word datasets show that this method is more superior to the state-of-the-art on rumor detection.

Key Terms: ***Graph Convolutional Networks(GCN), Rumor Detection, Recurrent Neural Network(RNN)***

I INTRODUCTION

A rumor is commonly defined as a statement whose truth value is unverifiable or deliberately false. These rumors spread on social media, carrying unreal or even malicious information, which will bring massive damage to individuals and society. With the development of social media and the popularity of mobile devices, it becomes increasingly easy to post rumors and spread rumors on social media. Widespread rumors may cause public panic and negative impact on individuals, which makes the automatic detection of rumors become necessary. Most of the existing rumor detection methods were based on statistical machine learning. Recently, some models for rumor detection exploited neural networks were inspired by the success of neural networks model in other tasks. This proposed model uses graph convolutional networks to capture user behaviour effectively for rumor detection. This model is composed of three modules: 1) auser encoder that models users attributes and behaviors based on graph convolutional networks to obtain user representation 2) a propagation tree encoder, which encodes the structure of the rumor propagation tree as a vector with bridging the content semantics and propagation clues; 3) an integrator that integrates the output of the above modules to identify rumors.

The growing influence experienced by the propaganda of fake news is now cause for concern for all walks of life. Election results are argued on some occasions to have been manipulated through the circulation of unfounded and doctored stories on social media including microblogs such as Twitter. All over the world, the growing influence of fake news is felt on daily basis from politics to education and financial markets. This has continually become a cause of concern for politicians and citizens alike. For example, on April 23rd 2013, the Twitter account of the news agency, Associated Press, which had almost 2 million followers at the time, was hacked. It would be interesting and indeed beneficial if the origin of messages could be verified and filtered where the fake messages were separated from authentic ones. The information that people listen to and share in social media is largely influenced by the social circles and relationships they form online.

II RELATED WORKS

Most previous approaches for rumor detection were based on statistical machine learning. These work designed a series of effective features in order to identify rumors. The designed features included three aspects: content-based, user-based and propagation-based. Content-based features consisted of length, symbols, sentiment, URLs, hashtag,, vocabulary and its part of speech. User-based features were statistics on registration age, friendship, activity level and history actions of users .Propagation-based features included the depth of the propagation tree, the number of comment, the number of retweet. Kumar A et al , [11] suggested an approach is based on long-short-term-memory and VGG-16 networks that show significant improvement in the performance, as evident from the validation result on seven different disaster-related datasets. Huang Qi et al, [8] discussed about the model that leverages graph convolutional networks to capture user behavior effectively for rumor detection. This work is composed of three modules: 1) a user encoder that models users attributes and behaviors based on graph convolutional networks to obtain user representation; 2) a propagation tree encoder, which encodes the structure of the rumor propagation tree as a vector with bridging the content semantics and propagation clues; 3) an integrator that integrates the output of the above modules to identify rumors. Traylor Terry et al, [13] explains the results of a fake news identification study that documents the performance of a fake news classifier are presented. The Textblob, Natural Language, and SciPy Toolkits were used to develop a novel fake news detector that uses quoted attribution in a Bayesian machine learning system as a key feature to estimate the likelihood that a news article is fake. Al-Ash et al, [1] reports on fake news and original news are represented according to the vector space model. Vector model combination of frequency term, inverse document frequency and frequency reversed with 10-fold cross validation using support vector machine algorithm classifier

III PROPOSED SYSTEM

Proposed model consists of three parts, namely user encoder, propagation tree structure encoder with content semantics and integrator. User encoder obtains the user representation using graph convolutional networks to model the user graph. Propagation tree structure encoder encodes the propagation tree by a tree-based recursive neural network. And then integrator combines these features to a fully connected layer for rumor detection.

In the user encoder module, try to consider more comprehensive and effective user information including user features and their behavioural information. In other words, In addition to the statistical-based user features used by most researchers, such as the number of followers, the number of fans and the age of registration, we want to capture the behaviour information. Further, It incorporate statistical-based user features and user behaviour information to get a high-level user representation in a more efficient and automated way. user encoder module, we use a graph convolutional networks, for the reason that graph convolutional networks have been proven to capture node features and graph structure features in graph structure data efficiently. A GCN is a multi-layered neural network that operates directly on the graph, which updates the representation of nodes based on the properties of their neighbourhoods.

In the propagation tree structure encoder module, It aim to capture the structural and semantic features of propagation tree. This system use a recursive neural network based on tree structure to capture propagation clues and semantic features. Since the traversal of tree structure data is directional, we adopt two different structures of recursive neural network: bottom-up RvNN encoder and top-down RvNN encoder.

The user encoder module obtains a user matrix that fuses user statistics features and behaviour information. And the propagation tree encoder module obtains a tree representation that combines the propagation tree structure and text semantic features. In order to fuse the output information of the two modules together, proposed system uses integration module in which user embedding combined with the propagation tree representations to produce a category prediction for each claim.

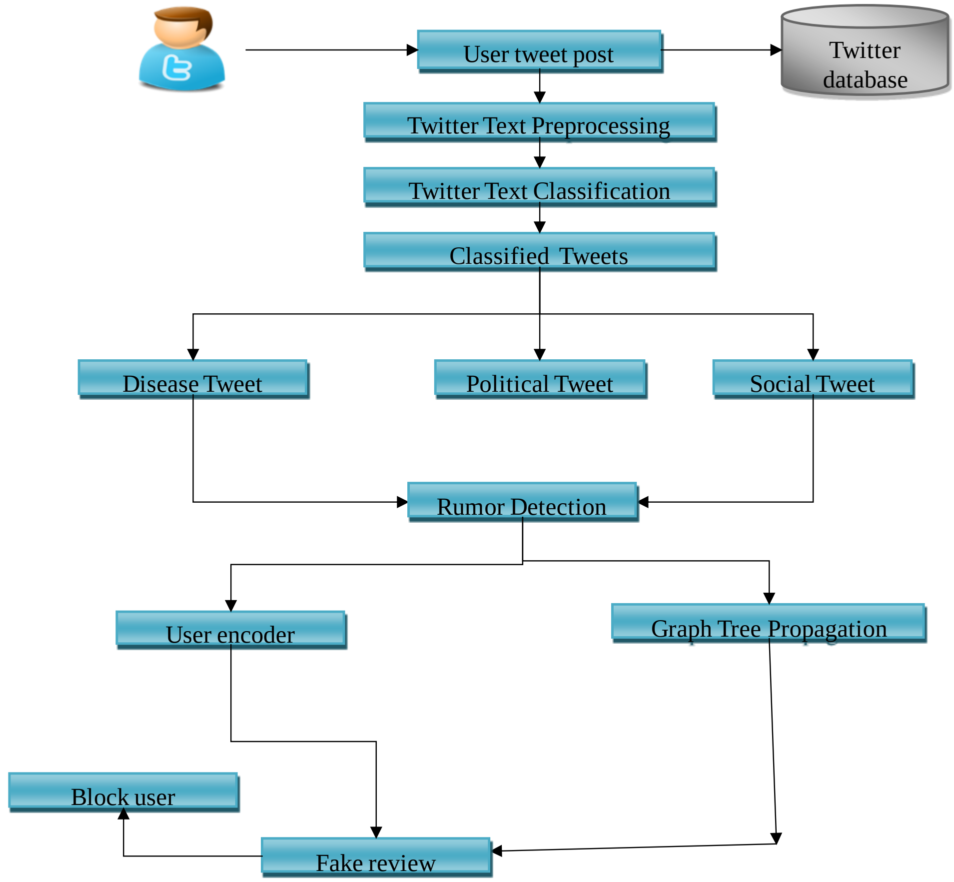

Fig 1 Architecture diagram for Proposed System

(i) Dataset Collection

This system use two publicly available twitter datasets namely Twitter15 and Twitter16. Twitter15 dataset contains 1,381 propagation trees and 276,663 users. Twitter16 contains 1,181 propagation trees and 173,487 users. Each propagation tree in the dataset is labelled as one of four categories, i.e., non-rumor, false rumors, true rumors, and uncertain rumors. Rumor detection dataset as a set engagements

Eg = {Eg1,Eg2,......,Egn}

where each engagement contains a claims set and a corresponding set of users, i.e., Egi = $\{C_i , U_i\}$, Ci represents all tweets that are ideally chronologically ordered by the reply source tweet ri, i.e, $C_i = \{ r , x_1,x_2 , x_3,\ldots\ldots.x_n \}$ where each xi_ is a responsive tweet of the root ri or the retweet of ri. Ui is the poster corresponding to each tweet in Ci.

According to the reply or repost relationship between the tweets, we can form Ci as a propagation tree structure with ri being the root node .

The goal of rumor detection task is to construct a classifier that can determine whether the claim in engagement is a rumor. The classifier can be formalized as a function f : Ci - Yi, where Yi is one of four categories: nonrumor,false rumor, true rumor, and unverified rumor.

(ii) Twitter text pre processing and classification

This component changes over crude content into effectively processable word tokens appropriate for utilize by our machine learning calculations. This system pre-process the content to:

1. Remove links, email addresses and mentions.

2. Translate html entities (e.g.   becomes the space character).

3. Translate emojis and emoticons into their name, according to a dictionary of well-known web emoji (emoticon, n.d.) and ASCII emoticons (gemoji, n.d.).

4. Quoted words are unquoted and prefixed with quote_ (For example *cough* and “cough” are replaced with quote_cough). This was implemented because words quoted in this way often denote sarcasm.

5. Hash tags are split into the individual words, by applying two different strategies: (a) For hashtags written in Camel-Case scripting notation, the words are split according to the case rules. (b) Otherwise a prefix-based space prediction algorithm is used to split the hashtag into the minimum possible number of words.

6. All punctuation and excess spaces are removed. Finally, text is converted to lower case.

This module performs automatic text classification of social media messages, using recurrent neural network It classify the tweet under the label such as politics, sports, health, cinema, tourism etc.

(iii) Rumor Detection : User Encoder –GCN

Rumor detection module contains three modules

i. user encoder

ii. Propagation tree encoder

iii. Integrator

User encoder module, use a graph convolutional networks, for the reason that graph convolutional networks have been proven to capture node features and graph structure features in graph structure data efficiently. A GCN is a multi-layered neural network that operates directly on the graph, which updates the representation of nodes based on the properties of their neighbourhoods. Graph convolutional networks have demonstrated its effectiveness in the node classification task: classifier with a GCN can learn the neighborhoods feature of nodes to provide information for node classification problems. Whether the GCN captures the information of the immediate neighbours (with one layers of convolution) or the neighbours information of the k-level hops (if K layers are stacked on top of each other) depends on how many layers of the convolution are used.

This module provides the user representation by capture the behavior information of users. Based on the co-occurrence relationship of users in different caims Ci, this module can form a user graph G = (V, E).

V - represents the node of the graph. The nodes are user who posts, reposts or replies to the tweets.

G - represents the edge of the graph. The coocurence relation derived from user behavioural information forms edges.

To formalize the edges of the graph, we introduce the adjacency matrix A .The values of the adjacency matrix A are defined as follows:

$$A_{ij} = \begin{cases} 1, \text{Link between user U1 and U2} \\ 0, \text{otherwise} \end{cases}$$

The features of all nodes in the graph can be defined as feature matrix X , The feature matrix X of these users is composed of user profile information, such as the number of fans, number of friends, registration details etc.

(iv) Propagation Tree Encoder-RNN

This module capture the structural and semantic features of propagation tree data. It uses the RNN to capture propagation clues and semantic features.

This module creates the the vector representation of the semantic text of root tweet. Based on the vector representation, this module classify all other reply tweet (child tweet) into two sets,

S1=(r,c1(r) – User who support and accept the root tweet)

S2=(r,c2(r) – User who oppose the content in the root tweet)

In the propagation tree structure encoder module, It aim to capture the structural and semantic features of propagation tree. This system use a recursive neural network based on tree structure to capture propagation clues and semantic features. Since the traversal of tree structure data is directional, we adopt two different structures of recursive neural network: bottom-up RvNN encoder and top-down RvNN encoder.

(v) Integrator

The user encoder module obtains a user matrix that fuses user statistics features and behavior information. And the propagation tree encoder module obtains a tree representation that combines the propagation tree structure and text semantic features. In order to fuse the output information of the two modules together, proposed system uses integration module in which user embedding combined with the propagation tree representations to produce a category prediction for each claim.

This module integrate the features obtained by user encoder and propagation tree encoder to produce the final category prediction(Rumor and non-rumor). It simultaneously learns the propagation structure information and user information to integrate the output of two modules (i.e) it reads the co-occurrence relation and feature matrix of the user who accept the content in root tweet.

Based on this, the integrator creates the trustworthiness value for user of root tweet. The final prediction is made by fed the trustworthy value into fully connected layer.

(vi) Blocking Rumor Spreading User

After the rumor get detected the fake user will be identified and intimated to the admin The admin will intimate three warnings to the user if he continues the user will be blocked permanently from the twitter system . If the news seems to be true the admin will allow the news to be spread if not the fake news will be automatically deleted

IV RESULT AND DISCUSSION

Performance analysis

In the experiments, proposed model basically achieved better performance than the other methods on the two datasets via capturing the user embedding with graph convolutional networks.

Statistic	Twitter 15	Twitter 16
Number of user	276663	173487
Number of source tweet	1490	818
Number of threads	331612	204820
Number of non rumors	374	205
Number of false rumors	370	205
Number of true rumors	372	203
Number of unverified rumors	374	203
Average length of tree	1337 Hours	848 Hours
Average number of post	223	251

Table 1 Statistic of the dataset

This system compares the performance of rumor detection with four baseline methods Decision Tree Classifier(DTR), Random Forest Classifier(RFC), Support Vector Machine-Hybrid Kernel(SVM-HK), BU-Hybrid and TD-Hybrid model.

This system has two sets of experiments: a first set of experiments on the multi-class problem ("true", "fake", "unverified", "non-rumor") and another set of experiments on the binary classification problem (retaining only news labeled "true" or "fake"). Reference papers usually carry out those two types of experiments

(i) Event Prediction
(ii) Rumor Detection

Event prediction is implemented with the RNN classification of training and testing approach can be added. The rumor detection further investigates the rumor spread from the root node using Graph tree propagation, user encoder and integrator.

It is observed that the performance of four baseline methods in the first group based on manual features are very poor, varying between 0.409 and 0.585 in accuracy. DTR ranks the candidate rumor cluster which collected related tweets matching a set of regular expressions to identify rumors. While only a few posts match these regular expressions, it performs the worst.

Method	Event Prediction Classification	Accuracy
Existing system	CNN	89.10
Proposed system	RNN	93.55

Table 2 Accuracy of classifying the Tweet Events

The implemented system well shown with the accuracy of the most forms which are all taken the total processing implementation are taken for the further identification systems.

Method	Precision	Recall	F-measure	Accuracy
RNN	0.832	0.999	0.9822	0.941
CNN	0.799	0.880	0.834	0.672

Table 3 Accuracy of fake news classification

Here, this model outperforms the current state of the art. Note that the size of this dataset is approximately half of the first one, so results (for every model) are subject to higher variance. It also induced higher overfitting on the validation set as can be seen on above table, a trend we didn't notice on the first dataset.

Deep learning models such as CNN and RNN often require much larger datasets, and in some cases multiple layers of neural networks for the effective training of their models. In our case, This system have a small dataset of 5,800 tweets. In our ongoing and future work we have collected the reactions of other users to these messages via the Twitter API in the magnitude of hundreds of thousands, with the aim of enriching the size of the training dataset and thus improving the robustness of the model performance. This model expect that this will also help to draw more actionable insights for the propagation of these messages from one user to another user.

V CONCLUSION

Automatic rumor detection that is able to identify rumors accurately and immediately from the social media. The proposed system present a hybrid neural network model for rumor detection on twitter.This work that model user with graph convolutional networks for rumor detection. The model consists of three modules: a user encoder module modelling the graph formed by user behaviors based on graph convolutional networks, a propagation tree structure encoder represents the propagation tree using a recursive neural network, and an integrator to integrate feature representations. This framework considers three aspects of rumor detection: contents, users, and propagation. This model covers influential factors of rumor detection : content, user, and propagation, and effectively captures information about those factors. User encoder models the user and propagation tree encoder learns the representation bridging the content semantics and propagation clues. Experiments on real-world datasets show that this model achieves much better performance than the state-of the-art method

REFERENCES

[1] Al-Ash, H. S., & Wibowo, W. C. (2018, July). Fake news identification characteristics using named entity recognition and phrase detection. In 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE) (pp. 12-17).IEEE.

[2] Aldwairi, M., & Alwahedi, A. (2018).Detecting fake news in social media networks. Procedia Computer Science, 141, 215-222.

[3] Buntain, C., & Golbeck, J. (2017, November).Automatically identifying fake news in popular Twitter threads.In 2017 IEEE International Conference on Smart Cloud (SmartCloud) (pp. 208-215).IEEE.

[4] Castillo, C., Mendoza, M., & Poblete, B. (2011, March).Information credibility on twitter. In Proceedings of the 20th international conference on World wide web (pp. 675-684).

[5] Chen, T., Li, X., Yin, H., & Zhang, J. (2018, June). 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, Cham.

[6] DiFonzo, N., & Bordia, P. (2007). Rumor psychology: Social and organizational approaches. American Psychological Association.

[7] Guo, H., Cao, J., Zhang, Y., Guo, J., & Li, J. (2018, October).Rumor detection with hierarchical social attention network.In Proceedings of the 27th ACM International Conference on Information and Knowledge Management (pp. 943-951).

[8] Huang, Q., Zhou, C., Wu, J., Wang, M., & Wang, B. (2019, July). Deep Structure Learning for Rumor Detection on Twitter. In 2019 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8).IEEE.

[9] Jin, Z., Cao, J., Guo, H., Zhang, Y., Wang, Y., & Luo, J. (2017, July).Detection and analysis of 2016 us presidential election related rumors on twitter.In International conference on social computing, behavioral-cultural modeling and prediction and behavior representation in modeling and simulation (pp. 14-24).Springer, Cham.

[10] Krishnan, S., & Chen, M. (2018, July). Identifying tweets with fake news. In 2018 IEEE International Conference on Information Reuse and Integration (IRI) (pp. 460-464).IEEE.

[11] 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.

[12] Kwon, S., Cha, M., Jung, K., Chen, W., & Wang, Y. (2013, December). Prominent features of rumor propagation in online social media. In 2013 IEEE 13th International Conference on Data Mining (pp. 1103-1108).IEEE..

[13] Traylor, T., Straub, J., & Snell, N. (2019, January).Classifying fake news articles using natural language processing to identify in-article attribution as a supervised learning estimator.In 2019 IEEE 13th International Conference on Semantic Computing (ICSC) (pp. 445-449).IEEE.