Rumor Detection on Hierarchical Attention Network with User and Sentiment Information

Sujun Dong[1], Zhong Qian[1(✉)], Peifeng Li[1,2], Xiaoxu Zhu[1], and Qiaoming Zhu[1,2]

[1] School of Computer Science and Technology, Soochow University, Suzhou, China
1820766741@qq.com, {qianzhong,pfli,xiaoxzhu,qmzhu}@suda.edu.cn
[2] AI Research Institute, Soochow University, Suzhou, China

Abstract. Social media has developed rapidly due to its openness and freedom, and people can post information on Internet anytime and anywhere. However, social media has also become the main way for rumors to spread largely and quickly. Hence, it has become a huge challenge to automatically detect rumors among such a huge amount of information. Currently, there are many neural network methods, which mainly considered text features but did not pay enough attention to user and sentiment information that are also useful clues for rumor detection. Therefore, this paper proposes a hierarchical attention network with user and sentiment information (HiAN-US) for rumor detection, which first uses the transformer encoder to learn the semantic information at both word-level and tweet-level, then integrates user and sentiment information via attention mechanism. Experiments on the Twitter15, Twitter16 and PHEME datasets show that our model is more effective than several state-of-the-art baselines.

Keywords: Rumor detection · User and sentiment information · Hierarchical attention network

1 Introduction

Rumors, usually used to spread panic and confusion, are untrue or inaccurate information breed on public platforms and Rumor Detection (RD) is to judge whether the information is true or false. Our work focuses on using relative public information to detect the false information spreading on social media. The key behind this work is that users on social media can express their opinions on the information disseminated on social media, and can provide evidence and speculation on false information [1].

In recent years, it has become increasingly popular to use neural network models to detect rumors. By modeling text information on social media, for example, Ma et al. [6–10] proposed a series of RNN-based methods, and these methods can automatically obtain a high-level text representation to detect the true degree of information. However, they only focused on how to use the text information of the rumor, and did not pay enough attention to user information, or even ignored it. Moreover, these methods hardly considered the role of sentiment information. Different users hold different degrees of credibility, and the sentiment expressed by them is directly related to their opinions.

X. Zhu et al. (Eds.): NLPCC 2020, LNAI 12431, pp. 366–377, 2020.
https://doi.org/10.1007/978-3-030-60457-8_30

Table 1. An example of a rumor source tweet and its user features

Source tweet	*Breaking! A four meter long Cobra with three heads was found in Myanmar!*
User Features	**username:** ABCD_1234 **verified:** False **description:** **follower:** 15 **listed_count:** 300 **user_creat_time:** 2011/10/4 9:36:17 **tweet_creat_time:** 2011/10/4 17:52:36 … …

Table 1 shows an example of a rumor source tweet and its user features. User *ABCD_1234* posted a tweet: *A 4* m *long cobra with three heads was found in Myanmar*. The tweet sounds appalling, but considering the characteristics of many snakes in Myanmar, it is difficult to identify *true* or *false* of the tweet, and may eventually be predicted as *Unverified*. However, combined with user information, the authenticity of this tweet can be predicted more accurately. First, in terms of user name, "*ABCD_1234*" is composed of "*ABCD*" and "*1234*" that is just a sequence of letters and digits without any actual meanings, indicating that this is probably not a normal user. Next, the user's verified property is *False*, showing that the user has not been verified who may spread false information maliciously. Finally, the user had no user description and posted 300 tweet in less than 24 h of user creations. Based on the above information, it is easy to determine that this is probably a user who specializes in spreading rumors, so the source tweet is identified as a rumor. Table 1 shows that user information plays an important role in RD.

In Table 2, the majority of users expressed fear of the source tweet, such as "*breaking*", "*terrify*", "*fear*", "*crying*", "*scared*" and so on, whose main purpose is to spread fear. Of course, there are users who express different sentiment information, such as "*fake*", "*don't believe*", etc. These sentimental information plays an important role in exposing rumors. Therefore, we think that considering sentiment information can effectively describe the sentiment features of a rumor in the process of spreading, so as to obtain more accurate high-level representation.

According to the above analysis, this paper proposes a novel hierarchical attention network with user and sentiment information (HiAN-US), which uses the hierarchical structure model to learn features from the word-level and tweet-level, respectively, and considers user and sentiment information via attention mechanism. Khoo et al. [1] showed that propagation tree structure on social media does not reflect the process of information propagation well. Therefore, the model uses linear structure to model text based on post time. In addition, the corresponding user and sentiment information are also used through word-level and tweet-level attention to improve the performance of rumor detection. The experimental results on Twitter15, Twitter16 and PHEME prove that our model HiAN-US is superior to the state-of-the-art baselines. The contributions of this paper can be summarized as the following two points:

Table 2. An example of a rumor source tweet and its reply tweets

Source tweet	*Breaking! A four meter long Cobra with three heads was found in Myanmar!*
Reply tweet	*Oh my god! It's terrify...*
	I'm fear! Crying...
	...
	Really? It is scared me!
	...
	Fake! I don't believe.
	It is fake! I know it at first sight.

- We propose a hierarchical model HiAN-US for RD to integrate the word-level and tweet-level information;
- Our model integrates user and sentiment information via attention at word-level and tweet-level.

The rest of the paper is organized as follows: Sect. 2 draws related research work in this field; Sect. 3 introduces the model proposed in the paper; Sect. 4 presents the experimental results and analysis; Sect. 5 summarizes the paper and proposes direction of future research work.

2 Related Work

Recently, rumor detection (RD) on social media has become a more popular topic, attracting more attention than before. Compared with the traditional machine learning methods, the neural network methods can more accurately learn the representation of text information, and does not require too many artificial labeling features. Therefore, more and more neural network methods are applied to rumor detection.

Chen et al. [2] proposed unsupervised rumor detection model based on user behavior, which combined RNN and AutoEncoder (AE) to obtain more features, in which user features encoded by RNN and then transmitted to AE. Do et al. [3] proposed a DRRD model based on text information and user information. The model used GRU to encode text information and user information, and then obtained a high-level text representation and user information representation through the maximum pooling layer. Finally, the two types of information were stitched together to predict whether the information to be tested is a rumor through the softmax classifier. Li et al. [5] used user information, attention mechanism and multi-task learning to detect rumors. After splicing user information and text information, they used LSTM to encode, and finally used the attention mechanism and softmax classifier to predict the true degree of information. Khoo et al. [1] proposed a rumor detection model named PLAN based on self-attention mechanism focusing on user interaction, in which only texts were considered, and user interaction was achieved by using self-attention mechanism. The above methods of using user information will be compared with the methods proposed by this paper:

- Compared with those studies that did not utilize self-attention in Transformer model (e.g., Chen et al. [2] and Do et al. [3]), our model uses a transformer encoder to encode user information. A large number of experiments prove that the transformer encoder is more effective than RNN. And our model uses attention to model user information, which highlights the impact of important users.
- Compared with those studies that only focused on text and user information (e.g., Li et al. [5]), first, the coding model we used is superior to LSTM; second, when using the attention mechanism, our method puts more emphasis on the role of users.
- Compared with those studies that only consider the propagation structure of tweets (e.g., Khoo et al. [1]), our model integrates user and sentiment information via attention, which is the main novelty of this paper.

As far as we know, previous work did not consider the effect of sentiment information for rumor detection. As shown in Table 2 above, sentiment information in tweets can also help to detect rumors. Therefore, we propose a hierarchical neural network with both user and sentiment attention for rumor detection.

3 Model for Rumor Detection

3.1 Problem Definition

We define each thread as $t_i = \{t_{i,1}, t_{i,2}, t_{i,3}, \ldots\ldots, t_{i,n}\}$, where $t_{i,1}$ is source tweet, $t_{i,j}$ is the j-th tweet in chronological order, and there are n tweets in the thread, whose user information and sentiment information is $u_i = \{u_{i,1}, u_{i,2}, u_{i,3}, \ldots\ldots, u_{i,n}\}$ and $s_i = \{s_{i,1}, s_{i,2}, s_{i,3}, \ldots\ldots, s_{i,n}\}$, respectively. The RD task is to assign a label y to each (t_i, u_i, s_i), where $y = \{False, True, Unverified, Non\text{-}Rumor\}$. In this paper, rumor means that a text may be true or false and should be identified furtherly. Therefore, *False* rumor means the text contains and spreads false information, while *True* rumor means the text contains true information, and *Non-Rumor* is true information without further identification. *Unverified* means it is difficult to judge whether the text is true or false due to the lack of the related information.

3.2 Hierarchical Attention Network with User and Sentiment Information

We propose a Hierarchical Attention Network with User and Sentiment information (HiAN-US) for RD and our model consists of five parts: word encoder layer, word attention layer, tweet encoder layer, tweet attention layer, and output layer. The encoder layers use transformer encoder to learn more semantic information on word-level and tweet-level, and use attention to integrate user and sentiment information between different levels. The architecture of our model is displayed in detail in Fig. 1.

Word Encoder Layer. We map each word of text information $(t_{i,j})$, user information $(u_{i,j})$ and sentiment information $(s_{i,j})$ in tweet to the vectors $\boldsymbol{w}_{i,j}^{k}$, $\boldsymbol{u}_{i,j}^{k}$, and $\boldsymbol{s}_{i,j}^{k}$, respectively. After stitching them together, they are fed to the word encoder layer for encoding. In

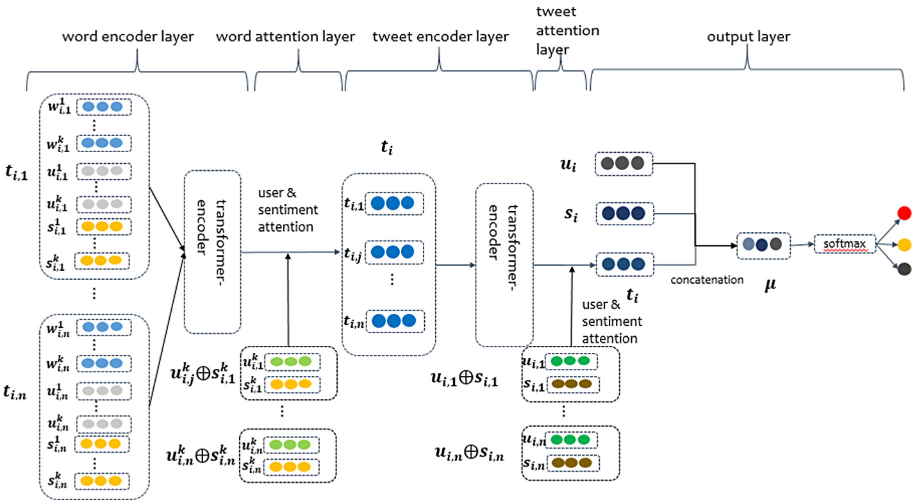

Fig. 1. Hierarchical attention network with user and sentiment information (HiAN-US)

this way, three types of information can be fused and encoded to improve the accuracy of the final text representation.

$$\boldsymbol{I}_{i,j}^{k} = \boldsymbol{w}_{i,j}^{k} \oplus \boldsymbol{u}_{i,j}^{k} \oplus \boldsymbol{s}_{i,j}^{k} \tag{1}$$

Attention Layer. In the sentiment analysis task, Yu et al. [11] have proved that the sentimental words in a text can express the author's sentiment to some extent. Therefore, we think that these sentimental words or sentences have an important role in improving the performance of rumor detection. Moreover, Li et al. [5] and Chen et al. [2] have proved that user features can significantly improve the performance of rumor detection. Hence, we introduce the attention mechanism based on user and sentiment information to our model at word-level and tweet-level, respectively.

1) Word-level Attention. We encode $\boldsymbol{I}_{i,j}^{k}$ through the word-level transformer encoder to obtain the vector $\boldsymbol{I}_{i,j}^{ka}$, and then feed it to the word-level attention based on user and sentiment information to obtain a high-level representation as follows.

$$\boldsymbol{m}_{i,j}^{k} = tanh\left(\boldsymbol{w}_{w}\boldsymbol{I}_{i,j}^{ka} + \boldsymbol{w}_{u}\boldsymbol{u}_{i,j}^{k} + \boldsymbol{w}_{s}\boldsymbol{s}_{i,j}^{k} + \boldsymbol{b}_{w}\right) \tag{2}$$

$$\boldsymbol{\alpha}_{i,j}^{k} = softmax\left(\boldsymbol{m}_{i,j}^{k}\right) \tag{3}$$

$$\boldsymbol{t}_{i,j} = \sum\nolimits_{k} \boldsymbol{\alpha}_{i,j}^{k}\boldsymbol{I}_{i,j}^{ka} \tag{4}$$

where $\boldsymbol{w}_{w}$, $\boldsymbol{w}_{u}$, $\boldsymbol{w}_{s}$, $\boldsymbol{b}_{w}$ are the parameters of the attention mechanism. $\boldsymbol{\alpha}_{i,j}^{k}$ indicates the importance of $I_{i,j}^{k}$ to $u_{i,j}^{k}$ and $s_{i,j}^{k}$. $\boldsymbol{t}_{i,j}$ is a high-level learned representation of a tweet.

2) Tweet-level Attention. After obtaining the word-level representation vector $\boldsymbol{t}_{i,j}$, the transformer encoder is also used to encode tweets to obtain the vector $\boldsymbol{t}_{i,j}^{a}$. Different sentiments play different roles in determining the true value of source tweets similarly, different users have different credibility for information disclosure. Therefore, similarly, a tweet-level attention mechanism based on user and sentiment information is adopted. The formula is expressed as follows.

$$\boldsymbol{m}_{i,j} = tanh\left(\boldsymbol{w}_t \boldsymbol{t}^{a}_{i,j} + \boldsymbol{w}_u \boldsymbol{u}_{i,j} + \boldsymbol{w}_s \boldsymbol{s}_{i,j} + \boldsymbol{b}_t\right) \tag{5}$$

$$\boldsymbol{\alpha}_{i,j} = softmax\left(\boldsymbol{m}_{i,j}\right) \tag{6}$$

$$\boldsymbol{t}_i = \sum\nolimits_j \boldsymbol{\alpha}_{i,j} \boldsymbol{t}^{a}_{i,j} \tag{7}$$

where $\boldsymbol{w}_t$, $\boldsymbol{w}_u$, $\boldsymbol{w}_s$, $\boldsymbol{b}_t$ are the parameters of the attention mechanism. $\boldsymbol{\alpha}_{i,j}$ indicates the importance of $t_{i,j}$ to $u_{i,j}$ and $s_{i,j}$. $\boldsymbol{t}_i$ is high-level representation of a set of tweets after learning.

Output Layer. The finally vector $\boldsymbol{\mu}$ consists of the vectors $\boldsymbol{t}_i$, $\boldsymbol{u}_i$ and $\boldsymbol{s}_i$ as follows.

$$\boldsymbol{\mu} = \boldsymbol{t}_i \oplus \boldsymbol{u}_i \oplus \boldsymbol{s}_i \tag{8}$$

where $\boldsymbol{u}_i$ represents the global user vector, which contains all the user information in t_i; $\boldsymbol{s}_i$ represents the global sentiment vector, which contains all the sentiment information in t_i . $\boldsymbol{u}_i$, $\boldsymbol{s}_i$ are obtained by average pooling of $\boldsymbol{u}_{i,j}$ and $\boldsymbol{s}_{i,j}$. $\boldsymbol{\mu}$ is the feature vector of a set of tweets, which contains the semantic information of tweets, the corresponding user and sentiment information of each tweet. After that, we will classify it through a softmax layer as follows.

$$\boldsymbol{p}(t_i) = softmax\left(\boldsymbol{w}_{\mu} \boldsymbol{\mu} + \boldsymbol{b}_{\mu}\right) \tag{9}$$

Finally, the loss function of our model is designed as follows.

$$L = -\sum_i y(t_i) \log(\boldsymbol{p}(t_i)) \tag{10}$$

where $y(t_i)$ is the ground truth, $\boldsymbol{p}(t_i)$ is the predicted probability of rumor for t_i.

4 Experimentation

This section details the datasets, data preprocessing, implementation details, and experimental results. We evaluate the model based on two datasets collected from social media. Experimental results prove that our model can achieve more satisfactory performance than several state-of-the-art baselines.

4.1 Experimental Setup

In this paper, two publicly available datasets, i.e., PHEME and Twitter (including Twitter15 and Twitter16), are used to evaluate our model. The PHEME is an expanded version of PHEME 5events and consists of 9 events. This dataset has two levels of annotation information: 1) annotation of thread as rumor or Non-Rumor; 2) comment rumor to true, false, or unverified. In Table 3, we give statistics on the data distribution of PHEME. For PHEME, our preprocess approach is different from Kumar and Carley et al. [4]: we divide the data randomly rather than based on events. For Twitter15 and Twitter16, we use the same processing methods as Khoo et al. [1]. The dataset annotates threads as true, false, unverified and Non-Rumor. Table 4 shows the data distribution of Twitter15 and Twitter16.

Table 3. Data distribution of PHEME

Events	Treads	Rumors	Non-Rumor	True	False	Unverified
Charlie Hebdo	2079	458	1621	193	116	149
Sydney Siege	1221	522	699	382	86	54
Ferguson	1143	284	859	10	8	266
Ottawa Shooting	890	470	420	329	72	69
Germany Crash	469	238	231	94	111	33
Putin Missing	238	126	112	0	9	117
Prince Toronto	238	229	4	0	222	7
Gurritt	138	61	77	59	0	2
Ebola Essien	14	14	0	0	14	0
Total	6425	2402	4023	1067	638	697

Table 4. Data distribution of Twitter15 and Twitter16

Dataset	Threads	True	False	Unverified	Non-Rumor
Twitter15	1413	350	334	358	371
Twitter16	756	189	172	190	275

According to the analysis on Twitter15 and Twitter16, we find that the proportion of retweeting source tweets in each thread is larger. We think that these retweets have little impact on RD and remove them. In the process of unified data processing, we replace all URLs with "URL". All "@XXX" and "#XXX" are divided into two words, i.e., "@nycaviation" is divided into "@" and "nycaviation".

Our model uses user and sentiment information. PHEME contains user information, while twitter15 and twitter16 not. So we only consider sentiment information for twitter15 and twitter16. The user and sentiment information used in PHEME are shown in Table 5.

Table 5. User and sentiment information

Information	Features
User	User name
	Verified
	Description
	Followers_count
	Follow_count
	Favorite_count
	Create_time
	List_count
Sentiment	Sentiment words
	Polarity(positive/negative)

4.2 Experimental Result

For the experiments on PHEME, Twitter15, and Twitter16, we used the same parameters while the word embeddings are initialized by GloVe. The following models are selected as the baselines.

- **RvNN** is a tree-based recursive neural network model proposed by Ma et al. [9].
- **PLAN** is one of the models proposed by Khoo et al. [1]. The model uses the maximum pooling layer at the word-level to obtain the tweet representation and the transform encoder to encode at tweet-level, and finally obtains a high-level representation of the entire thread through the attention.
- **HiAN-S** is our model, and only sentiment information is considered via attention at the word and tweet-level.
- **HiAN-U** is our model, only user information is considered via word-level and tweet-level attention.
- **HiAN-US** is our model, considering both user and sentiment information via attention at both word-level and tweet-level.

For the Twitter15 and Twitter16 datasets, we compare with RvNN and PLAN, while for the PHEME dataset, we compare with PLAN since RvNN only reported their results on the Twitter15 and Twitter16 datasets. Table 6 shows the results among different models on the three datasets.

For Twitter15 and Twitter16, as mentioned in 4.1, we only consider sentiment information to the model (HiAN-S). It can be seen from the experimental results that the performance of HiAN-S is higher than baseline systems on both datasets. On the Twitter15, the model is 11.3 higher than RvNN and 1.6 higher than PLAN on accuracy; on the Twitter16 dataset, the model is 8.5 higher than RvNN and 1.4 higher than PLAN on accuracy.

Table 6. Results of comparison with different methods

Dataset	Model	Accuracy
Twitter15	RvNN	72.3
	PLAN	82.0
	HiAN-S	**83.6**
Twitter16	RvNN	73.7
	PLAN	80.8
	HiAN-S	**82.2**
PHEME	PLAN	70.6
	HiAN-U	73.5
	HiAN-S	74.5
	HiAN-US	**77.7**

For PHEME, we can incorporate two information into the model: user information and sentiment information, which can produce three variants of the model: HiAN-U, HiAN-S, and HiAN-US. The experimental results show that the performance is significantly higher than the baseline system on this dataset, i.e., HiAN-U, HiAN-S and HiAN-US are 2.9, 3.9 and 7.1 higher than PLAN, respectively.

4.3 Analysis

Table 7 reports the experimental results of various models on four labels (FR, TR, UR, NR) on Twitter15, Twitter16, and PHEME. Form Table 7, we can find that our HiAN-S (with sentiment information) outperforms the other two baselines on NR on all three datasets, which shows that sentiment information is helpful to detect the tweets with NR. Our model HiAN-U (with user information) almost outperforms PLAN on three types except UR. This result further ensures the effectiveness of user information in RD. Combining the sentiment and user information into our model, HiAN_US can take the advantages from two aspects and further improve the performance. Especially, in comparison with HiAN-S and HiAN-U, HiAN-US improves the accuracies on FR and UR significant, with the gains of 6.0, 15.2, 8.8, and 13.1, respectively. This indicates that user information and sentiment information can complement each other to improve the performance of RD.

For different datasets, the performance of our model on each category is not always better than the baselines, but our model can achieve the best results on NR. We counted the average number of sentiment words contained in each tweet in different categories on each dataset as shown in Table 8. We can find that sentiment words contained in NR tweets of Twitter15, Twitter16 and PHEME are the majority, which are 4.83, 4.36 and 4.42 respectively. This explains why the performance of our model is higher than the baselines on NR. On Twitter15, the FR tweets contains an average of 4.28 sentiment words per tweet, which is relatively large, so the performance of HiAN-S in this category

Table 7. Performance on four types on Twitter15, Twitter16 and PHEME (FR: False Rumor; TR: True Rumor; UR: Unverified Rumor; NR: Non-Rumor)

Dataset	Model	Accuracy	FR	TR	UR	NR
Twitter15	RvNN	72.3	75.8	82.1	65.4	68.2
	PLAN	82.0	75.5	**90.2**	**82.2**	79.1
	HiAN-S	**83.6**	**83.3**	88.1	78.8	**84.2**
Twitter16	RvNN	73.7	74.3	83.5	70.8	66.2
	PLAN	80.8	**82.9**	88.4	77.5	72.7
	HiAN-S	**82.2**	77.6	**90.9**	**78.2**	**80.6**
PHEME	PLAN	70.6	56.3	62.8	62.7	77.1
	HiAN-U	73.5	57.6	**69.0**	52.4	81.8
	HiAN-S	74.5	60.4	63.3	50.3	84.9
	HiAN-US	**77.7**	**76.4**	58.9	**65.5**	**85.7**

can be higher than the baselines. On Twitter16, the FR tweets only contains an average of 1.60 sentiment words per tweet. Therefore, the sentiment information used by HiAN-S in this category is insufficient to improve performance.

Table 8. Statistics of sentiment words contained in each tweet

Dataset	FR	TR	UR	NR
Twitter15	4.28	4.03	2.70	4.83
Twitter16	1.60	2.67	2.71	4.36
PHEME	3.35	3.67	2.69	4.42

Table 9 shows an example of Non-Rumor in PHEME, while Table 10 shows an example of Non-Rumor user information in PHEME, including *username, verified, description, followers_count, list_count, follow_count*, etc. From the perspective of user information, source user is a verified news media with a large number of followers, relevant descriptions, and analysis of the information of reply users also shows that all users are in a normal state. From the perspective of sentiment information, source tweets do not contain sentiment words. There are panic words in reply users, such as "*awful*", "*very dark*", "*upsetting*", "*tragical*", etc., but there is no sentiment against source tweet. Our model can combine these two kinds of information to give source tweet and source user higher weight, so that it is easier to detect source tweet as Non-Rumor.

Based on the above analysis, the following conclusions can be drawn: 1) Our model can effectively improve the performance of Non-Rumor detection, thereby improve the overall performance; 2) Both user and sentiment information play an important role in

Table 9. An example of Non-Rumor in PHEME

Source tweet	• *french interior ministry: debris from #germanwings airbus a320 #4u9525 at 2,000m altitude http://t.co/8upmsinqkx http://t.co/miu94nhnbr*
Reply tweets	• *@skynews oh god.* • *@skynews a very dark day in the aviation industry .#germanwings* • *" @skynews: debris from #germanwings airbus a320 #4u9525 http://t.co/gzsw4yj6s2 http://t.co/sggsmujkly" awful, awful news. really upsetting* • *@skynews man this sucks. and i'm flying to barcelona in 2 weeks.* • *@leonhenry16 @skynews so? have u ever heard about such a tragical event happening twice within 2 weeks on the same route?*

Table 10. An example of Non-Rumor user information in PHEME

Source user	Sky news	True	True	1964051	15444	17
Reply user	Carly Marie	False	True	671	6	455
	Joseph Muiruri	False	True	540	20	1770
	Zia Lombardi	False	True	647	20	1362
	Leon Henry	False	True	12	0	66
	Screenwriting girl	False	True	1102	42	917

improving rumor detection performance; 3) User and sentiment information can form a complementary role to jointly improve the accuracy of rumor detection.

5 Conclusion

In this paper, a hierarchical attention network with user and sentiment information (HiAN-US) is proposed for RD. The model uses a transformer encoder to obtain semantic information at word-level and tweet-level. Different from previous research, we incorporate user information and sentiment information at both word-level and tweet-level via attention to capture more important components. Experiments on the PHEME, Twitter15 and Twitter16 datasets show that our proposed model is more effective than state-of-the-arts.

At present, most platforms can share messages in the form of text, pictures and short videos at the same time, and rumors can also be spread more quickly through these three ways. We consider how to combine three completely different information for multi-modal RD in future.

Acknowledgments. The authors would like to thank the three anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (No. 61772354, 61836007 and 61773276.), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

References

1. Khoo, L., Chieu, H., Qian, Z., Jiang, J.: Interpretable rumor detection in microblogs by attending to user interactions. In: 34th AAAI Conference on Artificial Intelligence, AAAI 2020, New York, pp. 8783–8790 (2020)
2. Chen, W., Zhang, Y., Yeo, C., Lee, B., Lau, C.: Unsupervised rumor detection based on users' behaviors using neural networks. Pattern Recogn. Lett. **105**, 226–233 (2018)
3. Do, T., Luo, X., Nguyen, D., Deligiannis, N.: Rumour detection via news propagation dynamics and user representation learning. In: IEEE Data Science Workshop, DSW 2019, Minneapolis, MN, USA, pp. 196–200 (2019)
4. Kumar, S., Carley, K.: Tree LSTMs with convolution units to predict stance and rumor veracity in social media conversations. In: 57th Conference of the Association for Computational Linguistics, Florence, pp. 5047–5058 (2019)
5. Li, Q., Zhang, Q., Si, L.: Rumor detection by exploiting user credibility information, attention and multi-task learning. In: 57th Conference of the Association for Computational Linguistics, Florence, pp. 1173–1179 (2019)
6. Ma, J., Gao, W., Wei, Z., et al.: Detect rumors using time series of social context information on microblogging websites. In: 24th ACM International Conference on Information and Knowledge Management, Melbourne, pp. 1751–1754 (2015)
7. Ma, J., Gao, W., Mitra, P., et al.: Detecting rumors from microblogs with recurrent neural networks. In: 25th International Joint Conference on Artificial Intelligence, New York, pp. 3818–3824 (2016)
8. Ma, J., Gao, W., Wong, K.F.: Detect rumors in microblog posts using propagation structure via kernel learning. In: 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, pp. 708–717 (2017)
9. Ma, J., Gao, W., Wong, K.-F.: Rumor detection on twitter with tree-structured recursive neural networks. In: The Meeting of the Association for Computational Linguistics, Australia, pp. 1980–1989 (2018)
10. Ma, J., Gao, W., Wong, K.F.: Detect rumor and stance jointly by neural multi-task learning. In: The Web Conference Companion, Lyon, pp. 585–593 (2018)
11. Yu, J., Jiang, J.: Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In: The 2016 Conference on Empirical Methods in Natural Language Processing, Austin, pp. 236–246 (2016)