Predicting Rumor Retweeting Behavior of Social Media Users in Public Emergencies

YONG TIAN[1], RONG FAN[2], XUEJUN DING[2], XIAXIA ZHANG[2], AND TIAN GAN[2]
[1]School of Physics and Electronic Technology, Liaoning Normal University, Dalian 116029, China
[2]School of Management Science and Engineering, Dongbei University of Finance and Economics, Dalian 116025, China

Corresponding author: Xuejun Ding (dingxuejun@dufe.edu.cn)

ABSTRACT Rumors in social media not only affect the health of online social networks, but also reduce the quality of information accessed by social media users. When emergencies occur, the rapid spread of rumors can even trigger mass anxiety and panic. However, the existing studies did not make a clear distinction between rumor and non-rumor information in public emergencies, so that they cannot effectively predict the rumor retweeting behavior. To this end, a model for predicting rumor retweeting behavior is presented based on the convolutional neural networks (CNN) called R-CNN model in the paper. In this model, the rumor retweeting behavior is considered as an important driving force of increasing the depth and breadth of rumor cascades, and four feature vectors are constructed with the historical textual content published by users, consisting of attention to public emergencies, attention to rumors, reaction time and tweeting frequency. To input the quantitative feature vectors for R-CNN, a K-means based core tweets extraction method is proposed to select the right tweets, and the quantitative feature representations are proposed. The predictive capability of the model has been proved by experiments base on two rumor datasets of emergencies crawled from Sina weibo. Experimental results indicate that the prediction accuracy of the model reaches 88%, and it can be improved by 7% on average compared with other models.

INDEX TERMS Behavior prediction, convolutional neural networks, public emergency, rumor.

I. INTRODUCTION

At present, in the period of social transition, China has been facing the severe challenges of various public emergencies, such as SARS in 2003, Songhua River water pollution in 2005, Illegal vaccines investigated in Shandong in 2016 and COVID-19 outbreak originating in Wuhan. These public emergencies not only disturb social order, but also can spark the spread of rumors which would create the negative emotions among people such as panic, anxiety and anger, and then it will induce people to have some irrational behavior, which affects the harmony and stability of the society [1].

Rumor is originally defined as the unproven narration or explanation concerning an event, and is related to objects, events or issues of public concern [2]. Some researchers further clarify the definition of rumor. Rosnow *et al.* think that rumors are fake conjecture about incident raised by personal anxiety [3]. Difonzo *et al.* point out that rumors are the

The associate editor coordinating the review of this manuscript and approving it for publication was Hocine Cherifi[ID].

information incongruous with the truth which can obscure the real situation [4]. Thus, it can be seen that rumor has two essential characteristics, that is, it is closely related to some specific events, and incongruous with the truth. Accordingly, in this paper, rumor is defined as the statement which is verified to be contrary to the fact, but to a certain extent, it is still spread by the public. The rapid spread of rumors may affect the public opinion of a social group, and thus have a negative impact on people's lives.

Recently, with the rapid development of Web 2.0 and mobile internet technology, Weibo, WeChat, Twitter and other mainstream social media have gradually become critical information sharing channels. However, social media just as a double-edged sword which helps users to get information more easily, at the same time, it also reduces the transmission cost of misinformation. Especially in the case of public emergencies, such as natural disasters, public health and social security events, the rapid spread of rumors in social media can easily cause mass anxiety and panic, which greatly tests the social governance capabilities and emergency response

capability of government departments [5]. Therefore, it is a challenging issue to reveal the spreading mechanism of rumors in social media under public emergencies, and to provide a scientific and effective rumor control strategies [6].

In recent years, researchers have made great efforts in the studies of rumor propagation issues from a macro perspective, e.g. spread time, diffusion mode, rumor content, and propagation process [7]. However, when social media users receive rumor messages, affected by rumor content attributes and individual behavior preference, they will make decisions whether to spread rumors or not. Therefore, individual behavior is an important internal motivation that affects the width and depth of rumor spreading cascade [8].

The current studies have shown that in the social system constructed by social media, individuals have dual identities of netizens and real social beings, so individual behavior has psychological and sociological characteristics [9], for instance, individual behavior is susceptible to the spiral of silence effect. Therefore, some researchers analyze the factors that affect individual spreading behavior in social media, and then establish a variety of models to predict individual spreading behavior in social media. Chen *et al.* propose a semi-supervised graph model (SGM) to predict the retweeting behavior by detecting users' emotional status corresponding with their current mood from their friends' tweets, then using Learn-to-Rank method, the Top-N retweets are obtained [10]. Ding and Tian build a model based on a back propagation neural network (BPNN) to predict the retweeting behavior of social media users, which extracts 11 feature vectors from recipient characteristics, retweeter characteristics, tweet content characteristics, and external media coverage [11]. Jiang *et al.* employ one-class collaborative filtering method to predict user's retweeting behavior by quantitatively measure the individual preference and social influence [12]. Zhang *et al.* propose a novel framework using probabilistic matrix factorization and collaborative filtering methods to predict retweeting behavior. The model established in their studies has a good prediction performance, which integrates the user-message retweeting data, social relationship contextual and message semantic embedding. Inspired by word2vec and co-factor matrix factorization model [13], Wang *et al.* propose a hybrid model, called HCFMF, which jointly decomposes the user-message matrix and message-message similarity matrix to predict individual retweeting behaviour [14].

The above studies have analyzed the different factors that affect information spreading behavior of social media users, and then established various models to predict users' retweeting behavior by introducing different features. The influence factors considered in existing studies can mainly be categorized into to the following two aspects.

(i) Content characteristics. Contextual features are the important factors to model users' retweeting behavior, such as text length, text features and users' information preference [15]. Furthermore, many empirical studies have confirmed that users' retweeting behavior is mainly affected by users' information preference and content of tweets [16].

(ii) Influence of external environment and neighbors on social media users. Under the spiral of silence effect, users' retweeting behavior can also be affected by external environment and their friends in social networks, and these external factors eventually turn into the psychological pressure on users. Due to these pressures, users are likely not retweet the topics which less retweeted by their neighbors in social media, although they are interested in these topics [17].

Although Existing studies have investigated the influence of various factors on users' retweeting behavior, they did not make a clear distinction between the attributes of public opinion information, that is, rumor information and non-rumor information. Studies have shown that compared with other information online, rumors in public emergencies are more likely to appear at the early stage of information diffusion and then accumulate in a short term; the sender and spreader of rumors may not be influential users, but relatively new users with highly dense or overlapping local networks; rumor cascades are often deeper than non-rumor cascades [18]. Moreover, the researchers investigate the propagation characteristics of rumor and non-rumor from the perspectives of information system [19], reliability of information [20], topic trend [21] and source of information [22], and they find that rumor and non-rumor are significant difference in generation mechanism, spreading channel and scale.

In light of the above analysis, individual retweeting behavior is an important internal motivation that affects rumor spreading mechanism [23], so under the influence of information content, social relationship and cyberspace environment, individual rumor spreading behavior is obviously different from the non-rumor spreading behavior. However, the existing prediction models cannot effectively predict the rumor retweeting behavior of social media users. Therefore, using the Convolutional Neural Networks (CNN), this paper constructs a rumor retweeting behavior prediction model R-CNN based on the user's historical tweets. The prediction performance of the model is verified by using the dataset crawled from Sina weibo.

The main contributions of this paper are summarized as follows.

(i) We propose a R-CNN model to predicting rumor retweeting behavior of social media users in public emergencies. In the model, we quantify the influence factors of rumors based on the historical textual content published by social media users, and use attention to public emergencies, attention to rumors, reaction time and tweeting frequency to predict rumor retweeting behavior of a user.

(ii) We propose a K-means based core tweets extraction method for selecting the right tweets to analyze the user's attention to public emergencies. To explore word embedding method, we propose a cosine similarity based method CSTF-IDF based on the TF-IDF method. Moreover, we propose the quantitative feature

representation of attention to rumors, reaction time and tweeting frequency.

(iii) We conduct a large number of simulation experiments to analyze the influence of the proposed quantitative features on predicting rumor retweeting behavior, and to verify the good performance of the proposed R-CNN model.

The rest of the paper is organized as follows. Section II describes the problem formalization of this paper. Section III proposes the R-CNN model for predicting rumor retweeting behavior. Section IV introduces the datasets, the measure values and the evaluation metrics using in the paper. Section V analyzes the quantitative features, and Section VI describes the experimental results and discussions. Finally, we give the conclusion in Section VII.

II. PROBLEM FORMALIZATION

Existing studies show that individual spreading behavior not only can be affected by some explicit factors, such as reaction time to events and retweeting frequency, but also affected by some implicit factors, such as individual psychological characteristics, individual interest and sensitivity to rumors [24]. Furthermore, according to the analysis that the attribute of public opinion is not explicit, most relevant studies assume that rumor and other information have no difference in spreading mechanism. However, the differences in attention to the information with different attributes will inevitably affect individual behavioral decision making [25]. In social media, users express their preference by tweeting or retweeting. The tweets can reflect the types of events users are interested in. It means that users' preference can be detected by what they tweet or retweet.

CNN [26] has been successfully applied to natural language processing. Compared with the bag-of-words (BoW) model which is the classic model of natural language processing, CNN do not have to train the topics of words and tweets. Thus, we can only quantify the sentence with itself instead of knowing much thing from social media. Moreover, unlike fully connected neural network, in CNN, each output is only connected to limited inputs which makes a decrease in computation scale when the input is multi-dimensional data, such as image, video and text. Another characteristic of CNN is weight sharing, which makes two adjacent values are highly correlated to form local feature [27].

Zhang *et al.* [26] propose a model based on CNN to predict individual retweeting behavior on Twitter. In their model, the text of users' historical tweets is taken as the input of the model and the word embeddings are quantified with the publicly available word2vec dictionary trained with Google news corpus. Considering the impact of public emergencies on psychology and behavior of social media users, as well as the differences in the spread characteristics of rumors and non-rumors, we introduce four quantitative features into the CNN consisting of attention to public emergencies, attention to rumors, reaction time and tweeting frequency. Then, we construct a new rumor retweeting behavior prediction model based on CNN, called R-CNN, which can reflect the influence factors of rumor retweeting behavior of social media users under public emergencies more comprehensively.

III. R-CNN MODEL FOR PREDICTING RUMOR RETWEETING BEHAVIOR

A. MODELLING
The overall structure of the proposed R-CNN model is shown in Fig.1.

In our model, a tweet is defined as a document and each document is segmented into multiple words. For encoding user's attention to public emergencies (*EA*) as a text feature at fully connected layer, we extract some core tweets of each user and use the word embedding method to convert each word of these tweets at word embedding layer. Then, all word embeddings are concatenated to build the tweet matrix of word embedding layer, as shown in formula (1). In the matrix, each column is a vector corresponding to a word and the sequence of vectors is the same with that of words in document. Each document can be encoded into a matrix of $64 \times n$ where n is the number of words in the document.

$$v_{doc} = \begin{bmatrix} | & | & & | \\ v_{w1} & ... & & v_{wn} \\ | & | & & | \end{bmatrix} \tag{1}$$

After that, we use the convolution layer to reduces the dimension of tweet matrix. First, we generate a filter matrix $w \in R^{l \times d}$, where l is the window size and d is the length of document. And b is a bias term. In our model, the number of filter matrixes and the window size l are alterable, and we define these two parameters as the filter size (*FS*) of R-CNN model. Second, we use formula (2) to calculate the inner product of word embeddings and filter matrix. In this step, the sliding step is set to 1 and each document is operated $64 - l + 1$ times.

$$vf_{wi} = g \left(w \cdot \begin{bmatrix} v_{wi} \\ \vdots \\ v_{w(i+l-1)} \end{bmatrix} + b \right) \tag{2}$$

In formula (2), g is the most commonly used non-linear activation function ReLu (Rectified Linear Unit) in CNN which can be represented as $f(x) = \max(0, x)$. The output of convolution layer is shown as formula (3). Then, the output of convolution layer is the input of pooling layer.

$$vf_{doc} = \begin{bmatrix} vf_{w1} \cdots vf_{w(n-l+1)} \end{bmatrix} \tag{3}$$

In pooling layer, we extract the maximum value for each vector with a max-overtime pooling operation for each filter matrix as formula (4), and we can construct the numbers of vectors of vF_{doc} by changing the filter size.

$$vF_{doc} = \max \begin{bmatrix} \max(vf_{w1}) \cdots \max(vf_{w(n-l+1)}) \end{bmatrix} \tag{4}$$

After the tweet matrix passes through the pooling layer, it is converted into a one-dimensional matrix, and then is input
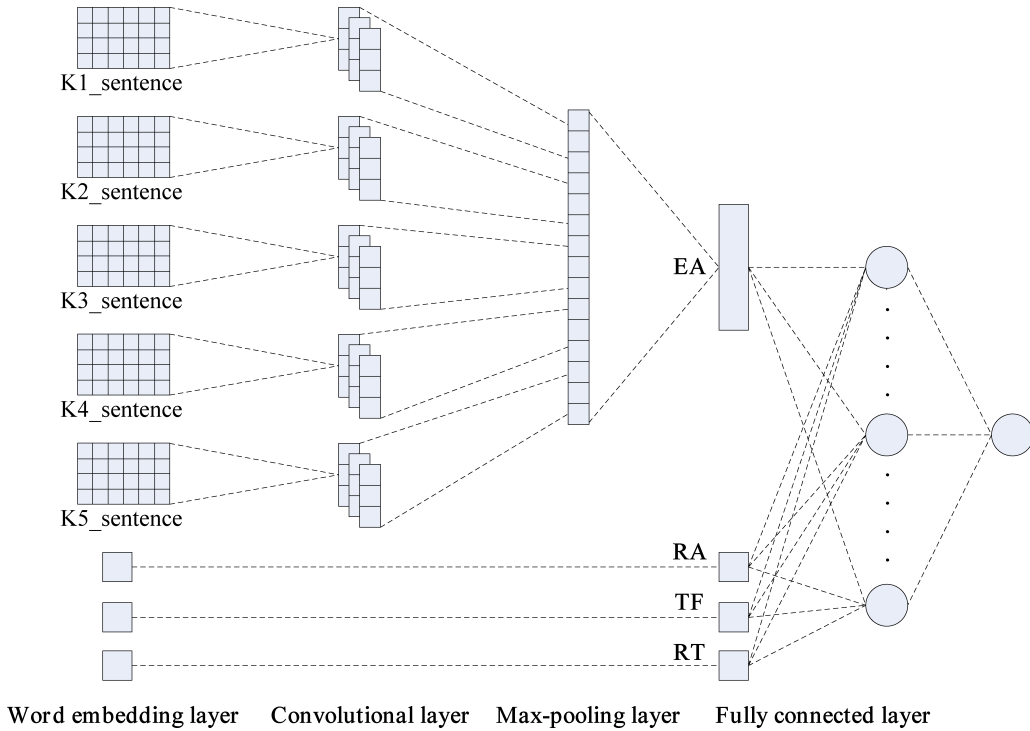
FIGURE 1. The structure of R-CNN.

into the full connection layer. At the same time, the attention to rumors (*RA*), reaction time (*RT*) and tweeting frequency (*TF*) of the user are directly input into the full connection layer from the word embedding layer.

In the parameter training step, we use the stochastic gradient descent (SGD) method to optimize the objective function. The pseudo code of the proposed R-CNN model is described as follows.

Algorithm 1

Input: All users' historical tweets; all users' information
Output: The result of predicting rumor retweeting behavior
  1: **for** each user **do**
  2:     Extract the core tweets with the Algorithm 2
  3:     Convert all word of core tweets with word2vec method
  4:     Build the tweet matrix v_{doc} with the formula (1)
  5:     Calculate vf_{wi} with the formula (2)
  6:     Construct the numbers of vectors of vF_{doc}
  7:     Quantify user's attention to rumors with formula (5)
  8:     Quantify user's reaction time with formula (6)
  9:     Quantify user's tweeting frequency with formula (7)
 10:    Optimize the objective function with SGD method
 11: **end for**

B. FEATURE EXTRACTION

Text feature is an important basis for predicting retweeting behavior. However, when the issue comes into rumor retweeting behavior prediction, only text feature is not enough to

TABLE 1. Feature vectors of R-CNN model.

Feature	Layers	Activation function	Optimizer
EA	Word embedding layer Convolutional layer Max-pooling layer Fully connected layer	ReLu	SGD
RA	Word embedding layer Fully connected layer	--	SGD
RT	Word embedding layer Fully connected layer	--	SGD
TF	Word embedding layer Fully connected layer	--	SGD

measure users' information preference. So, other features need to be extracted to evaluate users' attitude to rumors. Vosoughi *et al.* give a definition of tweet controversiality. In their paper, the more controversial of a tweet, the more likely that it is rumor [28]. That is to say, the more a user retweets controversial text, the more likely that he/she will retweet rumors in the future. Based on this finding, we can quantify a user's attention to rumors by analyzing the controversiality of his/her historical retweets. On the other hand, the time interval between the occurrence of public emergency and the first retweeting by user (i.e., reaction time), and user's own tweeting frequency also affect his/her retweeting behavior [29]. Therefore, we use the attention to public emergencies, attention to rumors, reaction time and tweeting frequency to model retweeting behavior of a user. The above feature vectors are described in Table 1.

C. FEATURE REPRESENTATION

1) ATTENTION TO PUBLIC EMERGENCIES

We use convolutional layer to encode a user's historical tweets and then generate the feature values of the user's attention to public emergencies at fully connected layer. Considering the difference in the number of historical tweets of different users, we cluster the tweets of each user into K clusters using K-means method, and then extract K core tweets of each user as the input of our model. The core tweets extraction method and the word embedding method are introduced in Section III-D and Section III-E, respectively.

2) ATTENTION TO RUMORS

It is extremely difficult to detect rumors before the truth comes. In social media, some users will comment on the unverified tweet before it is clarified, and the comments on a tweet reflect the controversiality of this tweet. The controversiality of a tweet is mainly presented with the following two aspects: (i) the number of users who comment on the tweet, i.e., the more users comment on a tweet, the more controversial it is; (ii) emotional properties marked as positive, negative and neutral, i.e., for a tweet, the closer the number of positive comments and negative comments, the more controversial it is. Based on the above analysis, a method to quantify user's attention to rumors is presented in formula (5), where RA_i is the ith user's attention to rumors, m is the number of historical retweets of this user, p is the number of positive comments on each historical retweet and n is the number of negative comments on each historical retweet.

$$RA_i = \frac{1}{m} \sum (p+n)^{\min(\frac{p}{n}, \frac{n}{p})} \qquad (5)$$

3) REACTION TIME

Social media users are sensitive to timeliness. That is to say, the more early a user accesses to the tweet, the more likely he/she will retweet it. In view of this, we define the time lag between posting the original tweet by the author and accessing to the tweet by a user as reaction time, as shown in formula (6), where PT_i is the time which the author posts the original tweet and UT_i is the time which a user accesses to the tweet.

$$RT_i = UT_i - PT_i \qquad (6)$$

4) TWEETING FREQUENCY

The frequency of using social media is one of the most important behavior characteristics of a social media user, which can be determined by two parameters: the number of user's historical tweets and the duration of using social media. So the tweeting frequency can be defined as formula (7), where $Ntweets_i$ is number of the ith user's historical tweets and $duration_i$ is the duration of the ith user using social media from the user registering on the social media platform.

$$TF_i = \frac{Ntweets_i}{duration_i} \qquad (7)$$

For convenience, we normalize each feature variable obtained by formula (5) to (7).

D. CORE TWEETS EXTRACTION METHOD

In social media, the number of historical tweets posted by each user is often large, and it also varies with the users. In addition, although a user has tweeted a lot, these historical tweets only focus on some specific preferences of this user generally. So, we cluster all historical tweets of each user with K-means method as follows: First, we only need to extract the user's attention to public emergencies from the user's tweets, so we don't need to pay too much attention to the impact of clustering number on tweet clustering accuracy. The existing study [26] shows that 5 clusters can reflect the preferences of social media users, so we set the clustering number to 5 in the paper, i.e., $K = 5$. Then, we define the centroid tweet of each cluster as core tweet.

The K-means method is a kind of sophisticated unsupervised learning algorithm which needs to perform multiple iterations. In this paper, considering that the text matrix is very sparse, instead of Euclidean distance, we compute a variant of cosine similarity as the distance of two tweets [30]. The cosine similarity uses the cosine value of the angle between two feature vectors to describe the similarity, which can reflect the relative difference. The cosine similarity has very good performance in measuring the text similarity, because it is very sensitive to text vectors. In our method, the cosine distance between tweet x and tweet y is defined as $d^c(x, y) = 1 - \cos(x, y)$, where $\cos(x, y)$ is the cosine similarity calculated as follows:

$$\cos(x, y) = \frac{\sum\limits_{k=1}^{n} x_k y_k}{\sqrt{\sum\limits_{k=1}^{n} x_k^2} \sqrt{\sum\limits_{k=1}^{n} y_k^2}}. \qquad (8)$$

At initialization time, a random tweet is selected as the first centroid; then the tweet with the largest distance from the previous generation centroid is selected as the next generation centroid until the method converges. The pseudo code of core tweets extraction method is described as follows.

E. WORD EMBEDDING METHOD

In order to seek a better way to embed words, we compare the performance of the classic word2vec [14], [15], [26] and TF-IDF [30] word embedding methods in the paper.

Word2vec is a neural network based word embedding method which can map a sparse one-hot word vector into a n-dimensional dense word vector. In word2vec method, a three layer neural network is constructed consisting of input layer, hidden layer and output layer, and a lot of words are used to train the neural network. The weights of the trained neural network are taken as the word vector.

Algorithm 2 Core Tweets Extraction Method

Input: All users' historical tweets; $K = 5$

Output: The centroid tweet of each cluster

1:   Random select a centroid γ_1

2:   **for** i in range $(K - 1)$ **do**

3:     **for** each tweet t_i **do**

4:       Calculate possibility of $t_i$to be next centroid with:
$$p_{ti} = (\tfrac{1}{2}d^2(t_i, \gamma_{best}))^2 \Big/ \sum (\tfrac{1}{2}d^2(t_i, \gamma_{best}))^2$$

5:     **end for**

6:   **end for**

7:   K centroids are obtained: $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5$

8:   **while** convergence condition is not satisfied **do**

9:     **for** each tweet t_i **do**

10:     Calculate its cluster with:$c^i = \arg\min_j(1 - \cos(x, y))$

11:    **end for**

12:    **for**each cluster j **do**

13:     Calculate possibility of t_i to be next centroid with:
$$p_{ti} = (\tfrac{1}{2}d^2(t_i, \gamma_j))^2 \Big/ \sum (\tfrac{1}{2}d^2(t_i, \gamma_j))^2$$

14:    **end for**

15:   **end while**

In TF-IDF method, each word embedding is represented by the TF-IDF value that can be calculated as follows:

$$tfidf_{ij} = tf_{ij} \cdot idf_i, \tag{9}$$

where $tfidf_{ij}$ represents the TF-IDF value of the ith word in the jth document, and tf_{ij} is the term frequency of the ith word in the jth document, and idf_i is the inverse document frequency of the ith word. The term frequency tf_{ij} of the ith word measures its importance in the jth document, and can be calculated as follows:

$$tf_{ij} = \frac{n_{ij}}{\sum_k n_{ij}}, \tag{10}$$

where n_{ij} is the number of times which the ith word appears in the jth document, and $\sum_k n_{ij}$ is the total number of words in the jth document. The inverse document frequency idf_i of the ith word measures its general importance in all documents, and can be calculated as follows:

$$idf_i = \lg \frac{|D|}{\left|\{j : t_i \in d_j\}\right|}, \tag{11}$$

where $|D|$ is the total number of documents, $\left|\{j : t_j \in d_j\}\right|$ is the number of documents that contain the ith word.

In the paper, each historical tweet is regarded as a document and each document is represented as a list of words. In tweets, synonyms are often used to express the same meaning, but TF-IDF method processes them as different words. Therefore, we propose an improved word embedding method using the cosine similarity based on TF-IDF method, namely CSTF-IDF. In order to solve the problem of variants and synonyms in Chinese texts, we introduce the HIT IR-lab Tongyici Cilin (Extended) dictionary to define the similarity degree of two words [31], i.e., when the similarity degree of

two words is greater than 0.5, we think that these two words embeddings are same.

IV. EXPERIMENTAL SETUP
A. DATA DESCRIPTION

The individual behavior data is usually difficult to obtain and quantify, so previous studies on individual behavior are mainly based on qualitative methods. Nowadays, a huge amount of data is distributed online, which provides significant opportunities to perform quantitative study on individual behavior. Sina weibo is an online media service for the Chinese community in China and around the world, and it is also one of the most influential social media platforms with the largest number of users in China. In view of this, from Sina weibo, we crawled the public opinion data on "The attack at Mizhi No.3 middle school" and "The incident of students falling from building in Chongqing", which are two typical public emergencies with great social impact in recent years in China. Then, two datasets of above two public emergencies are constructed respectively for predicting rumor retweeting behavior, as shown in Table 2.

TABLE 2. The datasets for rumor retweeting behavior prediction.

Item	Case 1	Case 2
Case description	The attack at Mizhi No.3 middle school[1]	The incident of students falling from building in Chongqing[2]
Rumor nodes	256	204
Non-rumor nodes	939	840
Number of historical tweets	284238	203523
Time Span of crawling data	2016.3.22 12:00~ 2016 3.30 23:59	2019.9.10 18:18~ 2019.9.17 23:59

In order to construct the feature vectors in the model, we crawled user data and textual content data related to the above two public emergencies from Sina weibo, respectively. The data crawled in our experiment includes the profile data of each user and 500 pieces of historical tweets before a user learnt about the event. In addition, we traced the source of these 500 historical tweets, and crawled the comments on each original tweet. Then, the data is cleaned by the following four steps:

(i) Cleaning text. For each historical tweet, we first remove its message header and URL, and then turn Emoji into words. Next, each tweet can be saved as a piece of data, which consists of user ID, tweet time and textual content. Finally, the historical tweets of all users are stored in an Excel file.

---

[1]It is reported by People's Daily that students from Mizhi no. 3 middle school were attacked by the suspects on their way home from school on 27 April 2018, which killed 7 students and injured 14 students.

[2]It is reported by China's Sohu.com that at a rainy night on 8 September 2019, 3 girls of 12 and 13-year-old fell from the 18th floor in a high-rise residential area in Shiqiao shop in Chongqing.

(ii) Processing user characteristic data. We extract the raw profile data of each user with number of followers, number of followees, number of history tweets, verification status (i.e., whether verified by the platform or not) and registration date. Then, these quantitative characteristic data are stored in an Excel file. It should be noted that, for convenience, we use the duration of a user using social media, which is defined as the time span from the user registering on the platform to the specific public emergency occurring, instead of the user's registration date.

(iii) Extracting user core tweet. The historical tweets of each user are encoded and clustered into 5 clusters using by the algorithm presented in section III-D, and then the centroid tweet of each cluster is extracted as a core tweet of the user.

(iv) Generating word vectors. To embed words, the publicly available word2vec dictionary of Chinese text is used in this paper. They are trained using by continuous BoW model with Chinese text from Sogou news, Baidu Baike and the text of fiction. The number of word2vec vectors is 6115353 and the dimension of each vector is 64.

(v) Analyzing emotional properties of tweets. The emotional properties of the comments on historical retweets are judged as positive, negative and neutral, then the user's attention to rumors can be calculated using by formula (5).

B. EXPERIMENT SETTINGS

The experimental system is set up using by TensorFlow, which is a large-scale machine learning system that operates in heterogeneous environments. In the experiment, we use the random gradient descent (SGD) method to optimize the algorithm, and use the two datasets in Table 2 as the experimental dataset. After data cleaning, 1195 samples of Case 1 and 1044 samples of Case 2 are left, which are defined as Dataset 1 and Dataset 2 respectively. For each dataset, 10% of samples are randomly selected as the training data, and the remaining 90% of the samples are the testing data. All experiments are conducted on an ordinary laptop computer with MacOS Sierra operating system, 1.4GHz Intel Core i5 CPU, 4G RAM and 128G storage space.

Comparing with the prediction value and actual value for each testing sample, four measure values are obtained:

(i) True Positive (*TP*), which is the number of positive samples predicted to be positive;

(ii) False Positive (*FP*), which is the number of positive samples predicted to be negative;

(iv) True Negative (*TN*), which is the number of negative samples predicted to be negative;

(v) False Negative (*FN*), which is the number of negative samples predicted to be positive.

Using by the four measure values above, three evaluation metrics, i.e. accuracy (*Acc*), recall (*R*), and F1-score (*F*1) can be obtained to evaluate the prediction performance of the R-CNN model, as shown in formula (12) to (15). Accuracy is defined as the proportion of samples correctly predicted, and recall is defined as the proportion of positive samples. F-value is the weighting-harmonic-mean value of accuracy

and recall, and in the experiment, we use the F1-score (i.e., in formula (15), $\alpha = 1$), which is widely used to evaluate the performance of a binary classifier, to measure the prediction performance of our model.

$$Acc = \frac{TP + TN}{TP + FN + FP + TN} \tag{12}$$

$$R = \frac{TP}{TP + FN} \tag{13}$$

$$F = \frac{(1 + \alpha) \times TP}{(1 + \alpha) \times TP + FP + FN} \tag{14}$$

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{15}$$

V. QUANTITATIVE FEATURES ANALYSIS

As described in section I, some empirical studies have shown that retweeting behavior is affected by user's characteristics in social media platform, such as the number of followees, the number of followers, and the duration of using social media platform. However, these studies did not make a clear distinction between rumor and non-rumor information. Therefore, in this section, we will analyze whether these above characteristics need to be introduced into R-CNN model so as to improve prediction accuracy. The experiments are conducted on Dataset 1.

TABLE 3. Correlation analysis of rumor retweeting behavior.

	Coefficients	Std	t Stat	P-value
gender	-0.8233	1.1865	-0.6939	0.4879
followee	0.0002	0.0016	0.1118	0.9110
TF	0.0003	3.4262E-05	8.8965	3.1976E-18***
follower	-0.0002	0.0002	-1.3594	0.1744
duration	0.0004	0.0008	0.5919	0.5541
RT	-0.0184	0.0098	-1.8853	0.0597*

A. CORRELATION ANALYSIS

In the experiment, we analyze the correlation between retweeting behavior (dependent variable) and the following quantitative features (independent variables): user gender (*Gender*), the number of followers (*follower*), the number of followees (*followee*), duration of using the social media platform (*duration*), tweeting frequency (*TF*) and reaction time (*RT*) between the occurrence of a public emergency and the first retweeting by user. The value of dependent variable (i.e., retweeting behavior) is "1" when the user retweet rumors, and it is "0" when the user does not retweet rumors. From the analysis results given in Table 3, it can be seen that there is a positive correlation between *TF* and user's retweeting behavior, and a negative correlation between *RT* and user's retweeting behavior.

---

³***, **, * represent the significant level of 1%, 5%, 10%, respectively.

B. PERFORMANCE EVALUATION OF THE MODEL WITH DIFFERENT FEATURE SETS

Furthermore, to investigate the prediction performance of the model introducing the above different feature sets, we design the following four experiments: Experiment EA+1, EA+2, EA+3 and EA+4. In these experiments, we introduce the additional quantitative features *follower*, *followee* and *duration* respectively which are not considered in R-CNN model. Then, we compare the prediction accuracy of the model when introducing same text features and different quantitative features.

1) EXPERIMENT EA+1

In Experiment EA+1, we input *EA* and one quantitative feature into model, and the alternative features include *RT*, *RA*, *follower*, *followee*, *TF*, and *duration*. Moreover, we predict the rumor retweeting behavior by using the R-CNN model without introducing any quantitative features, for convenience, the model is called NR-CNN model in this paper. As shown in Table 4, compared with the prediction accuracy of NR-CNN (81%), the prediction accuracy of the model introducing *followee* or *duration* becomes worse, and it becomes better when introduce other features. These findings indicate that the prediction performance of the model cannot be improved after introducing *followee* or *duration*. From Table 4, we also can seen that the prediction accuracy can be greatly improved when introducing *RT*, and this result reveals that the rumor retweeting behavior of social media users is affected by timelines of emergencies. The reason is that new things and mass emerging information on social media can easily divert users' attention from a specific event. Moreover, refutation information may also affect the decision of user spreading rumor [32]. For example, if a user has known and accepted the truth, he/she will not retweet the relevant rumor. So, the rumor retweeting behavior of social media users is significantly affected by *RT*.

TABLE 4. Prediction results of experiment EA+1.

Feature set	Acc
EA + RT	0.84
EA + RA	0.83
EA + follower	0.83
EA + TF	0.81
EA + followee	0.80
EA + duration	0.80

2) EXPERIMENT EA+2

In Experiment EA+2, we input *EA* and two quantitative features into the model. As shown in Table 5, compared with the prediction accuracy of NR-CNN (81%), the model can achieve better prediction performance with the following 8 feature sets: $EA + RA + RT$; $EA + RA + follower$;

TABLE 5. Prediction results of experiment EA+2.

Feature set	Acc
EA + RA + RT	0.86
EA + RA + follower	0.85
EA + RA + TF	0.83
EA + RA + followee	0.80
EA + RA + duration	0.80
EA + follower + RT	0.84
EA + follower + TF	0.82
EA + follower + followee	0.81
EA + follower + duration	0.80
EA + RT + TF	0.85
EA + RT + followee	0.81
EA + RT + duration	0.80
EA + TF + followee	0.80
EA + TF + duration	0.84
EA + followee + duration	0.82

$EA + RA + TF$; $EA + follower + RT$; $EA + follower + TF$; $EA + RT + TF$; $EA + TF + duration$; $EA + followee + duration$. From Table 5, we also can see that no matter what other features are introduced, as long as the feature set contains *followee* or *duration*, the prediction accuracy of the model will be reduced, which is consistent with the result of Experiment EA+1. Consequently, *followee* and *duration* need to be removed from the set of features.

3) EXPERIMENT EA+3

In Experiment EA+3, we input *EA* and three quantitative features into the model. Based on the above analysis, *followee* and *duration* are not considered in this experiment. As shown in Table 6, the model can achieve best prediction performance (88%) with the feature set of $EA + RA + TF + RT$. It is noted that the prediction accuracy may not be increased when introducing more quantitative features into the model. For example, the prediction accuracy of the model with the feature set of $EA + follower$ (83%) is higher than that with the feature set of $EA + follower + TF$ (82%), and when we add another quantitative feature *RT* into the feature set,

TABLE 6. Prediction results of experiment EA+3.

Feature set	Acc
EA + RA + TF + RT	0.88
EA + RA + follower + TF	0.84
EA + RA + follower + RT	0.81
EA + TF + RT + follower	0.80

i.e. *EA + TF + RT + follower*, the prediction accuracy is further reduced (80%) which is even lower than that of NR-CNN model (81%). A possible explanation for this is that the interaction between features may lead to the degradation of prediction performance.

4) EXPERIMENT EA+4

In Experiment EA+4, we investigate the prediction performance of the model when introducing the feature set of *EA + TF + RT + follower + RA*. Although the feature set is constructed with *EA* and all the alternative quantitative features, the prediction effect is still unsatisfactory (80%).

Next, we compare the highest prediction accuracy of each experiment in the above four experiments, as shown in Fig. 2. From Fig. 2, we can seen that in Experiment EA+3, the model can achieve the best prediction performance, and the corresponding feature set is *EA + TF + RA + RT*. Accordingly, we use the feature set *EA + TF + RA + RT* as the input of our model, which reflects the influence of individual characteristics of social media users and external environment factors on rumor retweeting behavior, and the prediction accuracy of the model we constructed is 4.67% higher than other models on average.
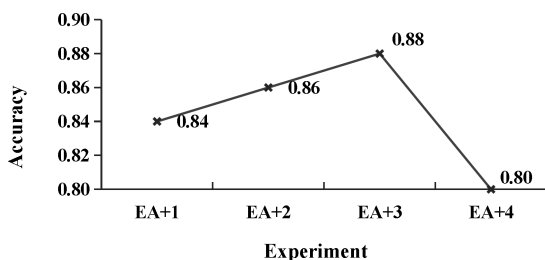

FIGURE 2. Prediction accuracy of the model in four experiments.

VI. EXPERIMENTAL RESULTS AND DISCUSSIONS
A. MODEL PARAMETER OPTIMIZATION
In order to achieve the best prediction performance, we optimize the parameters of the model with Dataset 1. In the convolutional layer, we adjust the filter size (i.e., the number of filter matrixes and the window size l), and for each filter, we set the number of feature maps to 100. The dropout rate p is set to 0.5. The mini-batch size of training process also can affect the prediction performance of the model, so in order to determine the optimal value of filter size (*FS*) and batch size (*BS*), we compared the prediction accuracy of the model with different values of the above two parameters, as shown in Table 7 and 8.

From above two tables, we can see that when the value of *FS* is fixed, the prediction accuracy of the model at *BS* = 32 is higher than that at *BS* = 16. However, when *BS* is fixed, compared with the case of the fixed *FS*, the prediction accuracy of the model has a greater fluctuation with the

TABLE 7. Prediction performance with different *FS* (*BS* = 32).

FS	Acc	R	F1
[1,2]	0.84	0.67	0.62
[1,2,3]	0.88	0.72	0.67
[2,3,4]	0.83	0.63	0.67
[2,3]	0.81	0.63	0.63
[3,4]	0.78	0.50	0.50

TABLE 8. Prediction performance with different *FS* (*BS* = 16).

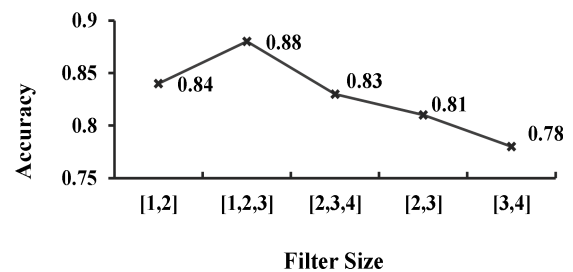| FS	Acc	R	F1
[1,2]	0.84	0.67	0.62
[1,2,3]	0.84	0.67	0.62
[2,3,4]	0.82	0.63	0.67
[2,3]	0.81	0.63	0.63
[3,4]	0.81	0.63	0.63

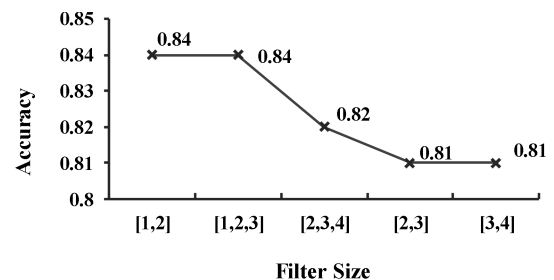**FIGURE 3.** The trend of accuracy under different FS (BS = 32).

FIGURE 4. The trend of accuracy under different FS (BS = 16).

change of *FS*, as shown in Fig. 3 and Fig. 4. In addition, the changes of F1-score and recall have little effect on the prediction accuracy of the model. Given this, in our model, *FS* and *BS* are set to [1], [2], [3] and 32 respectively.

Next, we adjust the number of epochs in the model, i.e. the training times, and find that as the number of epochs increases, the prediction accuracy tends to be stable, as shown in Fig. 5. From Fig. 5, it can be seen that the model can achieve the optimal prediction performance when the number of epochs reaches 200. So, we set the number of epochs to 200 in the model.
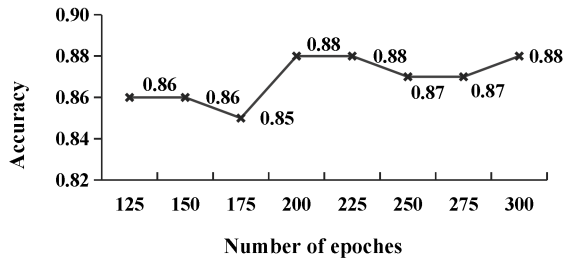
FIGURE 5. Prediction accuracy changes with the number of epochs.

TABLE 9. Prediction performance with dataset 1.

Word Embedding method	Acc
CSTF-IDF	0.84
Word2vec	0.88

TABLE 10. Prediction performance with dataset 2.

Word Embedding method	Acc
CSTF-IDF	0.80
Word2vec	0.86

TABLE 11. Prediction performance with dataset 1.

Model	Acc	R	F1
LR	0.74	0.44	0.47
NR-CNN	0.81	0.57	0.66
RF	0.81	0.66	0.57
DT	0.83	0.75	0.60
BPNN	0.83	0.63	0.67
SVM	0.84	0.77	0.69
R-CNN	0.88	0.72	0.67

TABLE 12. Prediction performance with dataset 2.

Model	Acc	R	F1
LR	0.62	0.44	0.42
NR-CNN	0.76	0.66	0.61
DT	0.78	0.62	0.58
RF	0.81	0.75	0.50
BPNN	0.82	0.66	0.57
SVM	0.82	0.75	0.50
R-CNN	0.86	0.71	0.71

B. EVALUATION OF WORD EMBEDDING METHOD

To evaluate the word embedding methods, we compare the prediction performance of R-CNN model which respectively uses word2vec and CSTF-IDF method. We set $BS = 32$ and $FS = [1, 2, 3]$. The results are shown as Table 9 and 10. From the Table 9 and 10, it can be seen that the word2vec method using trained dictionary can achieve the higher prediction accuracy for both two datasets. It is because that the word2vec method fully considers co-occurrence relationship of words and meanings of words. The CSTF-IDF method achieves the lower prediction accuracy for both two datasets because it is not good enough to process the relationship between words and meanings of words. Therefore, in the paper, we use the trained word2vec method to embed words. In this method, we use a large number of articles and news to train the dictionary so that the word embeddings are objective.

C. EVALUATION OF R-CNN MODEL

To evaluate the prediction performance of R-CNN model proposed in this paper, we conduct two comparative experiments. First, we compare the prediction performance of R-CNN model with that of the NR-CNN model. Then, based on the same dataset and feature vectors, we compare the prediction performance of R-CNN model with that of 5 classical machine learning models respectively, which are Support Vector Machine (SVM), Logistic Regression (LR), Random Forest (RF), Decision Tree (DT) and Back Propagation Neural Network (BPNN).

The parameters of the above two experiments are set as follows. (i) In NR-CNN and R-CNN, we set BS to 32 and FS

to [1], [2], [3]. (ii) We implement the five machine learning models (SVM, LR, RF, DT and BPNN) with sklearn package in python. In SVM, we choose RBF as the kernel function parameter. In BPNN, we build a neural network with three layers: input layer, hidden layer with three neurons and output layer with two neurons. In DT, we choose GINI as the optimization method, and set maximum feature parameter to 5 and maximum depth parameter to 6. In RF, we set minimum samples split parameter to 2. In LR, we choose saga as the optimization method. The prediction results of the above two experiments are shown as Table 11 and Table 12.

From Table 11 and 12, it can be seen that among all the seven models, R-CNN can achieve the highest prediction accuracy (88% for Dataset 1 and 86% for Dataset 2) which is 7% higher than other models on average, and LR has the worst performance (74% for Dataset 1 and 62% for Dataset 2). Compared with NR-CNN, the prediction accuracy of R-CNN is improved greatly after introducing quantitative features. It follows from this that individual characteristics and attention to rumors play an important role in predicting the retweeting behavior of social media users.

VII. CONCLUSION

In recent years, all kinds of public emergencies occur frequently all over the world. Meanwhile, social media, such as Twitter, Sina weibo and Facebook, has become the most important channel of public opinion diffusion. Different from the traditional mass media, social media has a huge number of

information sources, of which most are ordinary users called ''grassroots''. Consequently, when public emergencies occur, relevant public opinion information can spread very quickly and widely in social media, and it is easy to breed rumors which have negative impact on social stability. Therefore, it is of great practical significance to study the rumor spread mechanism in social media under public emergencies for formulating effective online rumor governance strategies.

In this paper, using by the data from Sina weibo, we analyze the influencing factors of rumor retweeting behavior of social media users. Considering that the special attributes of rumors, we find that in the context of public emergencies, the retweeting behavior of rumor is different from that of general information. Thus, the existing models cannot effectively predict the rumor retweeting behavior of social media users. A novel rumor retweeting behavior prediction model based on CNN, i.e. R-CNN, is proposed in this paper. In the model, we use four quantitative features, attention to public emergencies, attention to rumors, reaction time and tweeting frequency. The conclusions are as follows.

(i) The model can achieve good prediction performance: a) The prediction accuracy of the model reaches 88%; b) Compared with NR-CNN based on text features only, by introducing quantitative features, the prediction accuracy of the model can be improved by 7%. To further verify the prediction performance of R-CNN model, we compare the model with several classic behavior prediction models (LR, RF, DT, BPNN and SVM). The results show that the prediction accuracy of R-CNN is 7% higher than that of other models on average.

(ii) The experimental results in this paper also reveals that the behavior decision of rumor retweeting is determined by both internal factors and external factors. In terms of internal factors, retweeting behavior is mainly affected by attention to public emergencies, attention to rumors and tweeting frequency; in terms of external factors, rumor retweeting behavior is mainly affected by reaction time, that is to say, the earlier the rumor is exposed, the more likely the individual is to spread them. The experimental results also show that the prediction accuracy of the model is improved greatly when we add reaction time as the quantitative feature into the feature sets, and some individual characteristics, such as number of followers, number of followees and duration of using the social media platform, cannot improve the prediction performance.

The findings of this study are helpful for social media platforms and relevant departments to accurately predict the rumor retweeting behavior of social media users in the context of public emergencies, and effectively identify the possible rumor speakers, so as to develop scientific and reasonable rumor suppression measures. In the future, we will use some larger datasets to validate the performance and scalability of R-CNN model, and further improve the model by analyzing the influencing factors of users' rumor retweeting behavior from the perspective of individual psychological characteristics and emotional tendencies.

REFERENCES

[1] C. Fan, Y. Wang, and L. Huo, ''Rumor propagation model with holing type-II functional response,'' *J. Syst. Manage.*, vol. 26, no. 1, pp. 71–77, Jan. 2017.

[2] W. A. Peterson and N. P. Gist, ''Rumor and public opinion,'' *Amer. J. Sociol.*, vol. 57, no. 2, pp. 159–167, Sep. 1951.

[3] R. L. Rosnow, ''Inside rumor: A personal journey,'' *Amer. Psychologist*, vol. 46, no. 5, pp. 484–496, May 1991.

[4] N. Difonzo and P. Bordia, *Rumor Psychology: Social and Organizational Approaches*. Washington, DC, USA: American Psychological Association. 2007.

[5] J. Wang, P. Zhang, and L. Liu, ''On Internet rumor spread effectiveness evaluation,'' *J. Intell.*, vol. 35, no. 1, pp. 105–109, Jan. 2016.

[6] Z. Tan, J. Ning, Y. Liu, X. Wang, G. Yang, and W. Yang, ''ECRModel: An elastic collision-based rumor-propagation model in online social networks,'' *IEEE Access*, vol. 4, pp. 6105–6120, 2016.

[7] L. Huo, F. Ding, T. Lin, and S. Yao, ''Dynamical analysis of rumor spreading model with incubation mechanism and activity of nodes,'' *IEEE Access*, vol. 7, pp. 152493–152500, 2019.

[8] D. Lin, B. Ma, M. Jiang, N. Xiong, K. Lin, and D. Cao, ''Social network rumor diffusion predication based on equal responsibility game model,'' *IEEE Access*, vol. 7, pp. 4478–4486, 2019.

[9] V. Rubin, ''Deception detection and rumor debunking for social media,'' in *The SAGE Handbook of Social Media Research Methods*. London, U.K.: SAGE, 2016. [Online]. Available: https://uk.sagepub.com/en-gb/eur/the-sage-handbook-of-social-media-research-methods/book245370

[10] J. Chen, Y. Liu, and M. Zou, ''User emotion for modeling retweeting behaviors,'' *Neural Netw.*, vol. 96, pp. 11–21, Dec. 2017.

[11] X. Ding and Y. Tian, ''Predicting retweeting behavior based on BPNN in emergency incidents,'' *Asia Pac. J. Oper. Res.*, vol. 34, no. 1, Feb. 2017, Art. no. 1740011.

[12] B. Jiang, J. Liang, Y. Sha, R. Li, W. Liu, H. Ma, and L. Wang, ''Retweeting behavior prediction based on one-class collaborative filtering in social networks,'' in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Pisa, Italy, 2016, pp. 977–980.

[13] K. Zhang, X. Yun, J. Liang, X.-y. Zhang, C. Li, and B. Tian, ''Retweeting behavior prediction using probabilistic matrix factorization,'' in *Proc. IEEE Symp. Comput. Commun. (ISCC)*, Messina, Italy, Jun. 2016, pp. 1–8.

[14] C. Wang, Q. Li, L. Wang, and D. Dajun Zeng, ''Incorporating message embedding into co-factor matrix factorization for retweeting prediction,'' in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Anchorage, AK, USA, May 2017, pp. 1265–1272.

[15] Z. Wang, Y. Guo, J. Wang, Z. Li, and M. Tang, ''Rumor events detection from Chinese microblogs via sentiments enhancement,'' *IEEE Access*, vol. 7, pp. 103000–103018, 2019.

[16] J. Shi, K. K. Lai, P. Hu, and G. Chen, ''Understanding and predicting individual retweeting behavior: Receiver perspectives,'' *Appl. Soft Comput.*, vol. 60, pp. 844–857, Nov. 2017.

[17] J. Zhang, J. Tang, J. Li, Y. Liu, and C. Xing, ''Who influenced you? Predicting retweet via social influence locality,'' *ACM Trans. Knowl. Discov. Data*, vol. 9, no. 3, pp. 1–26, Apr. 2015.

[18] S. Kumar and N. Shah, ''False information on Web and social media: A survey,'' 2018, *arXiv:1804.08559*. [Online]. Available: http://arxiv.org/abs/1804.08559

[19] F. Darabi Sahneh, C. Scoglio, and P. Van Mieghem, ''Generalized epidemic mean-field model for spreading processes over multilayer complex networks,'' *IEEE/ACM Trans. Netw.*, vol. 21, no. 5, pp. 1609–1620, Oct. 2013.

[20] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, ''FakeNewsNet: A data repository with news content, social context and spatialtemporal information for studying fake news on social media,'' 2018, *arXiv:1809.01286*. [Online]. Available: http://arxiv.org/abs/1809.01286

[21] L. Zhang, J. Zhao, and K. Xu, ''Who creates trends in online social media: The crowd or opinion leaders?'' *J. Comput.-Mediated Commun.*, vol. 21, no. 1, pp. 1–16, Jan. 2016.

[22] R. Drezewski, J. Sepielak, and W. Filipkowski, ''The application of social network analysis algorithms in a system supporting money laundering detection,'' *Inf. Sci.*, vol. 295, pp. 18–32, Feb. 2015.

[23] W. Jong and M. L. A. Dückers, ''Self-correcting mechanisms and echo-effects in social media: An analysis of the 'gunman in the newsroom' crisis,'' *Comput. Hum. Behav.*, vol. 59, pp. 334–341, Jun. 2016.

[24] M. Imran, C. Castillo, and F. Diaz, ''Processing social media messages in mass emergency: A survey,'' *ACM Comput. Surv.*, vol. 47, no. 4, pp. 67.1–67.38, Jul. 2015.

[25] Y. Zhou, C. Wu, Q. Zhu, Y. Xiang, and S. W. Loke, "Rumor source detection in networks based on the SEIR model," *IEEE Access*, vol. 7, pp. 45240–45258, 2019.

[26] Q. Zhang, Y. Gong, J. Wu, H. Huang, and X. Huang, "Retweet prediction with attention-based deep neural network," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, Indianapolis, IN, USA, Oct. 2016, pp. 75–84.

[27] M. Al-Sarem, W. Boulila, M. Al-Harby, J. Qadir, and A. Alsaeedi, "Deep learning-based rumor detection on microblogging platforms: A systematic review," *IEEE Access*, vol. 7, pp. 152788–152812, 2019.

[28] J. Li, N. Wu, and Z. Feng, "Model-based non-Gaussian interest topic distribution for user retweeting in social networks," *Neurocomputing*, vol. 278, no. 22, pp. 87–98, Sep. 2017.

[29] S. Vosoughi, M. Mohsenvand, and D. Roy, "Rumor gauge: Predicting the veracity of rumors on Twitter," *ACM Trans. Knowl. Discov. Data*, vol. 11, no. 4, pp. 1–36, Jul. 2017.

[30] B. Wang, W. Liu, and X. Hu, "Research on text clustering for selecting initial cluster center based on Cosine distance," *Comput. Eng. Appl.*, vol. 54, no. 10, pp. 11–18, Oct. 2018.

[31] L. Lv, W. Liang, and S. Ran, "A method for measuring word similarity based on Cilin," *Modern Comput.*, no. 1, pp. 3–6, Jan. 2013.

[32] Y. Tian and X. Ding, "Rumor spreading model with considering debunking behavior in emergencies," *Appl. Math. Comput.*, vol. 363, Dec. 2019, Art. no. 124599.

YONG TIAN received the B.S. degree in applied electronic technology from the Wuhan University of Technology, Wuhan, China, in 1998, and the M.S. degree in signal and information processing and the Ph.D. degree in micro-electronics and solid state electronics from the Dalian University of Technology, Dalian, China, in 2005 and 2014, respectively. He was a Visiting Scholar with Shanghai Jiao Tong University, Shanghai, China, from 2017 to 2018. He is currently an Associate Professor with the School of Physics and Electronic Technology, Liaoning Normal University, Dalian. His research interests include wireless networks, big data, and embedded systems.

RONG FAN received the B.S. degree in information management and information system and the M.S. degree in management science and engineering from the Dongbei University of Finance and Economics, Dalian, China, in 2017 and 2020, respectively. Her research interests include network opinion and machine learning.

XUEJUN DING received the B.S. degree in electronic information engineering and the M.S. degree in signal and information processing from the Dalian University of Technology, Dalian, China, in 2001 and 2005, respectively, and the Ph.D. degree in technical economy and management from the Dongbei University of Finance and Economics, Dalian, in 2014. She was a Visiting Scholar with the University at Buffalo, Buffalo, USA, from 2017 to 2018. She is currently an Associate Professor with the School of Management Science and Engineering, Dongbei University of Finance and Economics. Her research interests include wireless networks, management information systems, and big data.

XIAXIA ZHANG received the B.S. degree in marketing management from Hebei Agricultural University, Baoding, China, in 2018. She is currently pursuing the master's degree with the Dongbei University of Finance and Economics, Dalian, China. Her research interest includes network rumor.

TIAN GAN is currently pursuing the B.S. degree in information management and information system and the joint B.S. degree in finance with the Dongbei University of Finance and Economics, Dalian, China. Her research interests include text emotion analysis and machine learning.

• • •