Hindawi
Mobile Information Systems
Volume 2022, Article ID 6725840, 8 pages
https://doi.org/10.1155/2022/6725840

Research Article

Network Rumor Detection Method Using Deep Learning in Big Data Environment

Xi Lei

Department of Artificial Intelligence and Big Data, Chongqing Industry Polytechnic College, Chongqing 401120, China

Correspondence should be addressed to Xi Lei; leixi@cqipc.edu.cn

Received 23 February 2022; Revised 14 April 2022; Accepted 20 April 2022; Published 9 May 2022

Academic Editor: Mukhtaj khan

Aiming to achieve efficient and accurate network rumor detection, this paper proposes a rumor detection method based on deep learning network. First, this method uses API interface and web crawler to construct a large data set of information samples on the microblog platform. Then, this method processes and analyzes the large data set through multiple embedded layers to provide a complete and reliable analysis data set for the detection model. The rumor detection model is composed of bidirectional long- and short-term memory (Bi-LSTM) network and convolutional neural network (CNN), which can realize deep and efficient feature extraction for the analysis data set and ensure the optimal performance of rumor detection method. The simulation results show that the precision and accuracy of the proposed method are 0.9771 and 0.9726, respectively, which are better than the comparison algorithms. The proposed method has effective network rumor screening and identification performance.

1. Introduction

In recent years, with the rise of social media, more and more people tend to get information on social media. Social media has now surpassed traditional media as the main news source [1–3]. This change may be due to the characteristics of social media: (1) news on social media is usually more timely than traditional news media such as newspapers or television [4]; (2) users can further share with friends on social media, and it becomes easy to discuss news [5].

However, it should also be noted that due to the complexity of information on the Internet, supervision is difficult. Because some social media platforms lack information review systems [6], it creates fertile soil for the emergence and spread of rumors. These rumors contain unconfirmed or even false information, which may cause public panic, cause serious economic losses, and have a negative impact on society [7].

According to the statistics in China New Media Development Report (2013), more than one-third of the 100 popular events on microblog in 2012 were false. At the same time, the report said that more than 59% of users will use microblog as the preferred platform for publishing information and spreading news [8]. Therefore, the comments on microblog are mixed and easy to be used by criminals to threaten social security.

At present, rumors on social media have become a serious concern, and researchers have made great contributions on how to detect rumors and try to eliminate their negative effects [9–11].

In most of the existing researches on rumor detection, researchers regard rumor detection task as a classification task, and it is usually a two classification task, that is, to judge whether a message or an event is "rumor" or "nonrumor" [12, 13]. At present, the method based on machine learning has begun to take effect. Reference [14] proposed a dual-supervised machine learning framework to detect rumors by filtering and then analyzing their language attributes. Reference [15] took support vector machine as a binary classification technology and compared it with social media content and news media to realize rumor detection and identification. However, researchers need to have a sufficient understanding of the scene and devote a lot of effort to designing distinguishing features. The designed features often have limitations, which may be limited to some specific scenes and do not have good generalization performance.

With the development of deep learning methods, the method based on deep neural network has become the research focus in the field of rumor detection. Compared with traditional machine learning methods, the deep learning model tends to rely less on complex feature engineering. It uses the multidimensional information of events and combines various source features to predict whether a message is a rumor [16–18]. Reference [19] proposed a convolution neural network method based on deterministic factors, which effectively classified events into rumors by using the inherent features of information sets. Reference [20] judged the rumor of the Cantonese sample set on Twitter based on the two-way gated cyclic unit network with attention mechanism. Reference [21] used convolution neural network and sentence embedding technology to determine whether the network information is correct or not. However, the above methods have few levels of network information sample processing and analysis, and only use a single embedded layer to process the sample data, which lacks a certain reliability of the training sample data. In addition, the feature extraction of the sample data set by the detection model is inaccurate, which makes it difficult to ensure the precision and accuracy of rumor recognition.

Based on the above analysis, aiming at the low identification ability of current network rumor detection methods, a network rumor detection and analysis method based on fusion depth network model is proposed. In this method, the sentence segmentation embedding layer and the position embedding layer are introduced into the embedding layer, which is cascaded with the word embedding layer to realize the reliable processing of the sample data set. Through Bi-LSTM and CNN combined network model, the data features of the tested sample set are continuously extracted for high-performance network rumor identification and analysis.

The rest of this paper is arranged as follows: the second section introduces the construction process of the data set used. The third section introduces the rumor detection model based on deep learning network. In Section 4, the proposed method is verified by experiments. The fifth section is the conclusion.

2. Data Set Construction

In the rumor detection model, the data set to be used is timely and effective [22]. Microblog is currently the social media platform with top traffic in China, with more than 556 million user resources and an average daily consumption of more than 33 billion per day. Therefore, this paper uses microblog data set to realize rumor detection research, in which crawler or API technology is used to grab the data set on microblog.

At present, there are two methods to collect microblog data; one is the API interface exposed through the microblog open platform, and the other is the crawler-based crawling method.

2.1. API-Based Method. The microblog open platform provides a public API interface. Users first register as developer users on the microblog open platform and obtain the permission to call the interface after review.

Developers can obtain relevant information about communication features, such as returning the latest forwarded microblog of an original microblog, obtaining comment content in batch, and obtaining the relevant information of user characteristics, such as the number of fans, followers, and microblogs of the user. The obtained information is returned in JSON format, and the developer can analyze it to get the corresponding information [23]. The APIs provided by microblog open platform are sorted as shown in Table 1.

Although developers can use public API interfaces, the microblog open platform limits the number of visits of developers, which brings difficulties to collection. At the same time, although the API collection method is relatively simple and easy to implement, the microblog limits the number of search contents. Generally, it will only display the search contents within one page and adopts the shielding or nondisplay strategy for other relevant information. Therefore, the data collection method using API interface is difficult to meet the real-time and large-scale collection of microblogs.

2.2. Web Crawler Method. Web crawler is a kind of network robot, which can crawl the page content according to certain rules [24].

Firstly, the content of microblog can only be accessed after the user logs in, so the crawler must simulate the process of user login to obtain page information. Secondly, microblog has restrictions on the frequency of users accessing the page in unit time. Generally, users or IP access the page only a limited number of times and will not visit a website frequently in a short time. For users with abnormal access, microblog will adopt the strategy to close account. Therefore, for the above two cases, this paper provides two optimization methods: antiblocking login strategy and antiblocking task scheduling strategy.

Figure 1 shows the data collection framework based on web crawler.

2.2.1. Antiblocking Login. The collection of microblog data requires account login. In order to prevent a single user from being targeted, it is necessary to prepare a large number of user IPs and assign the tasks to multiple accounts for alternating collection, so as to alleviate the collection pressure and overcome the search restrictions.

2.2.2. Antiblocking Task Scheduling. A user has limited access to the page in a short time and will not visit the page frequently. Therefore, distributed task scheduling can be adopted to distribute the page search task to different crawlers, so as to ensure that the collection system can search and collect different pages continuously and stably.

Table 1: API interfaces.

Read interface	Statuses/home_timeline	Get the latest microblog of the current login user and the users they follow
	Statuses/user_timeline	Get the microblog published by the user
	Statuses/repost_timeline	Return the latest forwarded microblog of an original microblog
	Statuses/mentions	Get the latest microblog of @ current user
	Statuses/show	Obtain single microblog information according to ID
	Statuses/court	Get the number of forwarded comments of the specified microblog in batch
	Statuses/go	Jump to a single microblog page according to ID
	Emotions	Get official expression
Write interface	Statuses/share	Third party sharing link to microblog

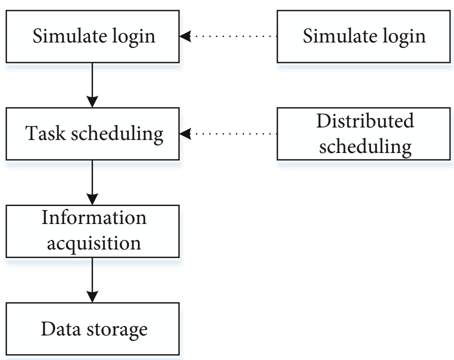

Figure 1: Microblog data collection framework.

The optimization measures of web crawler based on antiblocking login and antiblocking task scheduling can solve the problem that single IP collection is targeted and frequently visited pages are blocked.

3. Rumor Detection Model Based on Deep Learning Network

In this paper, the detection and analysis of network rumors are realized based on the improved convolutional neural network framework. The rumor detection problem is generally regarded as a classification problem. In the existing rumor detection methods, the shallow machine learning algorithm is mainly used, and the network structure of learning is represented by features, which is difficult to realize the in-depth extraction of the data features of the sample set, and can not ensure the effective and accurate detection of network rumor [25].

The rumor detection method proposed in this paper combines the deep memory network and convolutional neural network. On the premise of ensuring the deep network model, it can learn longer time series information through the complex gating structure, screen and extract the sample feature information, and complete the rumor detection task.

3.1. Problem Description. Generally speaking, the ultimate purpose of research on rumor detection is to predict which information is true, which is rumor or false information in a timely and accurate manner in many unmarked information. Therefore, the rumor detection task in this paper actually belongs to a binary classification task; that is, the model can predict the authenticity of the unmarked test data after training on the input data with label "rumor" or "non-rumor" [26]. In this paper, the formal definition of rumor detection is as follows.

First, a labeled training data set is given:

$$U = \{(m_1, n_1), (m_2, n_2), (m_3, n_3), (m_4, n_4), \cdots\cdots, (m_c, n_c)\}, \tag{1}$$

where c indicates the size of the data set; that is, the data set U contains a total of c records. The form of each record is $(m_i, n_i)(i = 1, 2, 3, \cdots\cdots, c)$, in which, t_1 indicates the text information recorded in U, $n_i \in \{t_1, t_2\}$ indicates the label of m_i, and the label values t_1 and t_2 indicate the category of "rumor" and "non-rumor", respectively.

For the network rumor detection task in this paper, after training a given data set U, a binary classification model is obtained: $m_i \longrightarrow t_j (j = 1\text{or}2)$, represented as the mapping from text m_i to a label value t_1 or t_2. The input data of the network rumor detection model is the text information to be judged, and the output of the model is the category label corresponding to the input text data, that is, whether the judged result is "rumor" or "non-rumor."

3.2. Overall Structure of the Model. The rumor detection method proposed in this paper combines the bidirectional long- and short-term memory network and convolutional neural network, proposes an improved deep learning network architecture Bi-LSTM-CNN for feature extraction, and then completes the classification and prediction of rumor through a combination of feed-forward neural network and Softmax normalized output layer.

In order to realize the early detection of rumors, each piece of data in the labeled training data set used is composed of text information and the label corresponding to the information. The input data of the model is only the content of the information itself and does not contain other relevant information such as the publisher of the information. The output of the model is the category label value of the predicted input text [27].

The overall architecture of the rumor detection model proposed in this paper is shown in Figure 2. According to

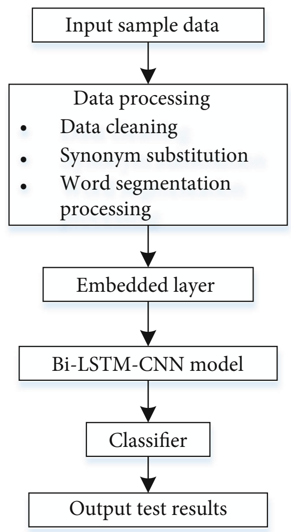

Figure 2: Network rumor detection architecture based on deep network.

the internal operation process of the model, the model can be roughly divided into four parts.

The first part is the processing of data text. Firstly, the input text is cleaned; then, the data is enhanced by means of synonym replacement, and finally, the text is segmented.

The second part is the embedding layer, which changes each marker symbol after word segmentation into the input vector representation of the converter.

The third part is the Bi-LSTM-CNN module, which is mainly used to extract text features and predict text categories.

The fourth part is classified output, which judges the category of input information through the predicted value.

3.3. Embedded Layer. The rumor detection model based on Bi-LSTM-CNN is a sentence-level binary classification model. Its input is a linear sequence, and the input data content is a single sentence text.

Different from other models, this paper adds two special embedding layers on the basis of word embedding layer, namely, sentence segmentation embedding layer and position embedding layer. Finally, the three vector representations obtained in the three embedded layers will simply sum the elements to obtain the data input of the detection model.

3.3.1. Word Embedding Layer. The number of hidden layer units in the rumor detection model in this paper is 698. Taking the input text in Figure 2 as an example, after word segmentation, the sentence will become 15 marker symbols. Therefore, if the example sentence in Figure 2 passes through the word embedding layer, each marker symbol in the sentence will be converted into a 698 dimensional vector. In other words, this sentence is converted into a matrix with a size of (15, 698). If batch size is included, it will be converted into a tensor with a size of (batch size, 15, 698).

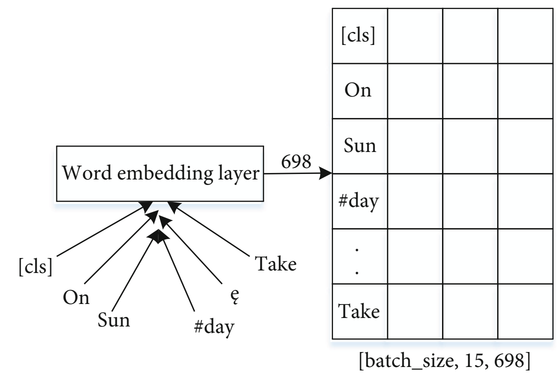

Figure 3: Output results of word embedding layer.

The output result of the example sentence after passing through the word embedding layer is shown in Figure 3.

3.3.2. Sentence Segmentation Embedding Layer. The main function of the sentence segmentation embedding layer is to distinguish two sentences in an input sentence pair. There are only two vector representations of 0 and 1 in the sentence segmentation embedding layer. The former vector assigns 0 to each marker in the first sentence, and the latter vector assigns 1 to each marker in the second sentence. The input of the rumor detection model in this paper is a single sentence text, so after the sentence segmentation and embedding layer processing, each sentence vector is 0.

The output result of the example sentence after sentence segmentation and embedding layer is shown in Figure 4.

3.3.3. Location Embedded Layer. In a sentence sequence, if the same word appears in different positions, the semantic information contained in the word may be different. Therefore, if a corresponding position vector is added to each word in the sequence, it can help the model better understand the order of input text.

The position vector in the position embedding layer is obtained through learning, and the maximum sequence length that the model can support processing is 256 marker symbols. Therefore, the position embedding layer here is essentially an embedded lookup table with the size of (256, 698), and the vector representation of words at the same position in different sequences is the same; that is, the first row of the table is the vector representation of any word in the first position, the second row is the vector representation of any word in the second position, and so on.

The output result of the example sentence after passing through the position embedding layer is shown in Figure 5.

After processed by the above three embedding layers, the word vector V_{terms}, text vector V_{text}, and position vector V_{position} are obtained in each layer. These three vectors are summed by adding elements to obtain a single vector V with a size of (batch_size, 15, 698).

The calculation formula of V is shown in the following formula.

$$V = V_{\text{terms}} + V_{\text{text}} + V_{\text{position}}. \tag{2}$$

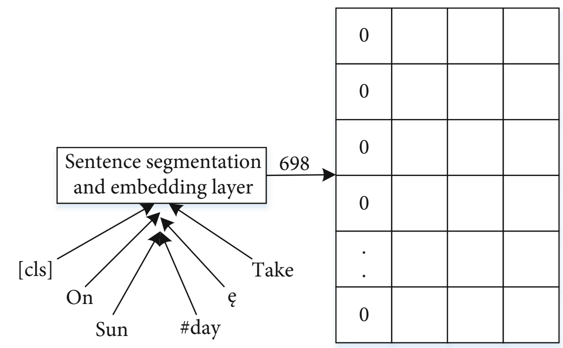

Figure 4: Output result of sentence segmentation embedding layer.

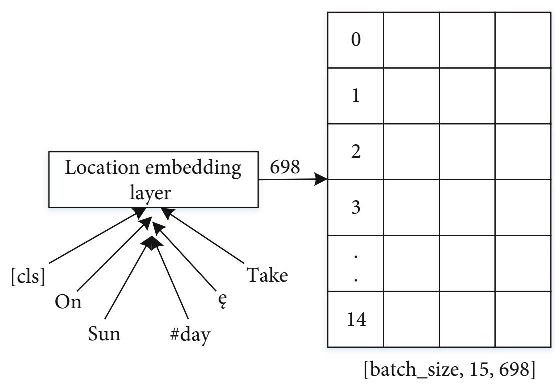

Figure 5: Processing results of position embedding processing layer.

This vector representation is the input vector of the bidirectional conversion encoder in the next rumor detection model.

3.4. Rumors Detection Based on Bi-LSTM-CNN. Although the convolution neural network can easily extract the local information in the text through the convolution operation, the performance of the sequence information for the text is not outstanding. Continuous sequence data such as text and voice usually use recurrent neural network to learn context information.

The long and short memory model, as one of the classical model in recurrent neural network, can learn longer time series information through complex gating structure and can also screen these information, which has strong feature extraction ability for sequence data. Therefore, this paper combines Bi-LSTM with CNN to extract text features, and the bidirectional recurrent neural network structure adds the influence of subsequent sequence information to the learning task. The short text classification algorithm based on Bi-LSTM-CNN model is shown in Figure 6.

As can be seen from Figure 6, the first step of the algorithm is the embedding layer. The words in the sentence are replaced by vectors through the look-up table operation, so as to obtain the vector representation of the text. The input of the model is $F = \{f_1, \cdots\cdots, f_{i+1}, \cdots\cdots, f_n\}$, where

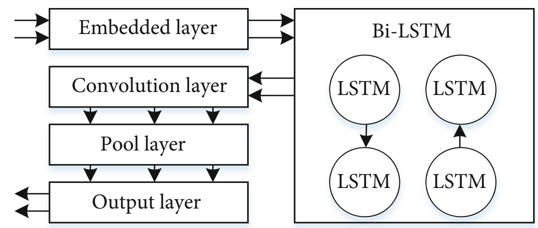

Figure 6: Rumor detection algorithm based on Bi-LSTM-CNN model.

f_i represents the i-th word in the text sentence and h_i represents the word vector corresponding to the word f_i in the vocabulary. Then, the sentence vector H is obtained through the word embedding layer, as shown as follows:

$$\begin{aligned} H &= \{h_0, \cdots\cdots, h_i, \cdots\cdots, h_{n-1}\} \\ &= Em(f_1, \cdots\cdots, f_{i+1}, \cdots\cdots, f_n). \end{aligned} \tag{3}$$

In the second step, each word vector in the sentence vector H passes through the forward LSTM model and the reverse LSTM model in turn, and the two results are combined and input into the next layer of neural network. The internal operation of Bi-LSTM has been introduced in the relevant technologies in Section 2. This step can be described by the following formula:

$$H = \left[\overrightarrow{\text{LSTM}}(H), \overleftarrow{\text{LSTM}}(H)\right], \tag{4}$$

where H represents the feature information extracted from the sentence vector after processing by Bi-LSTM model and H is a two-dimensional plane vector.

In the third step, the extracted feature information H further extracts local features through convolution neural network, which passes through convolution layer and pooling layer. The specific convolution operation and pooling operation are also described in detail in the relevant technologies in Section 2. Here, formulas (5) and (6) are used to describe these processes.

$$C_i = \text{Conv}(w_i, H), \tag{5}$$

$$P_i = \text{Pooling}(r_i), \tag{6}$$

where C_i is the convolution result of feature vectors H using convolution kernel w_i and P_i is the pooling result of C_i. After pooling, the features also need to be concatenate, which is expressed by a one-dimensional vector P.

Finally, the output of the pooling layer P is put into the fully connected layer based on the Softmax function to output the probability that the text data belongs to a certain category, as shown as follows:

$$A_g = \text{soft max}\left(w_g P + q_g\right), \tag{7}$$

where w_g represents the linear parameter of the fully connected layer, q_g represents the offset, and A_g represents the probability that the text belongs to a certain category, that is, the final output of the model.

4. Experiments and Result Discussion

In order to explore the performance of each rumor detection method, experiments were carried out on WeChat rumor data set. In order to obtain a model with stronger generalization ability, this paper divides all data set into training set, validation set, and test set according to the ratio of 8 : 1 : 1.

In the training stage, in order to make the training more stable, the learning rate is set to 0.0015. The model optimization method uses the random gradient descent algorithm and uses 100 texts as a batch for training. Each round uses all training texts and trains 350 rounds.

4.1. Model Evaluation Index. In order to evaluate the effectiveness of each rumor detection algorithm, four indicators are used here: precision Pre, recall Re, F_1, and accuracy Acc.

$$\begin{aligned} \mathrm{Pre} &= \frac{|T_P|}{|T_P| + |F_P|}, \\ \mathrm{Re} &= \frac{|T_P|}{|T_P| + |F_N|}, \\ F_1 &= 2 \times \frac{\mathrm{Pre} \times \mathrm{Re}}{\mathrm{Pre} + \mathrm{Re}}, \\ \mathrm{Acc} &= \frac{|T_P| + |T_N|}{|T_P| + |F_P| + |T_N| + |F_N|}, \end{aligned} \tag{8}$$

where T_N is the true negative rate, indicating the number of real events determined as real events; F_P is the false positive rate, which indicates the number of events that are actually real events but are judged as rumor events; F_N is a false negative rate, indicating the number of events that are actually rumored events but are judged to be true events; and T_P is the true positive rate, which indicates the number of rumor events determined as rumor events.

Specifically, the precision Pre is the proportion of rumor events among all events judged as rumor. However, in general, the number of positive samples in the data set is far less than the number of negative samples; that is, the positive and negative samples are unbalanced. The model can easily achieve high precision by making all predictions negative. Therefore, the recall Re is used to represent the proportion of all rumor events judged as rumor, which is used to evaluate the sensitivity of the model. F_1 is the harmonic average of the first two, indicating the comprehensive detection performance of the rumor detection method. Accuracy measures the similarity between events judged as rumors and true rumors. Please note that for these four indicators, the higher the value, the better the detection performance.

4.2. Sensitivity Analysis of Detection Model. TensorFlow framework is used to build various models in detection and analysis experiments. It is one of the mainstream deep learning model frameworks at present. In the improved embedding layer, the dimension of word vector is 350, so the dimension of hidden layer parameter vector in Bi-LSTM model is also set to 350.

In Bi-LSTM-CNN model, the different size of convolution kernel affects the size of each feature information, so the size of convolution kernel needs to be optimized according to the effect. Similarly, the number of convolution kernels will affect the number of channels after convolution, so the number of convolution kernels also needs to be optimized.

In this paper, the size and number of convolution kernels are optimized, and the average F_1 value of text classification on the validation set of the improved model is used as the index to find out a set of parameters with the highest classification accuracy as the final parameters of the model. The influence of different convolution kernel parameters on the model is shown in Table 2.

As shown in Table 2(a), when the size combination of convolution kernel is (4,6,8), the classification accuracy of the model is the highest on the validation set, and the index is 0.8775, indicating that the effect of the model is the best at this time. Therefore, in the Bi-LSTM-CNN model, the size combination of convolution kernel is set to (4,6,8).

At the same time, as can be seen from Table 2(b), when the number of convolution kernels is set to 256, the classification accuracy of the model on the validation set is the highest, and the evaluation index value is 0.8511, which is higher than the performance of the model when the convolution kernel is 512. Therefore, in the rumor detection model, the number of convolution kernels is set to 256.

4.3. Performance Analysis of Detection Model. This paper introduces reference [15] and reference [21] as comparative experiments. All detection run in the same environment to confirm the performance optimization of the proposed network rumor detection model based on deep learning network.

Figure 7 shows the rumor detection results of different detection methods for the same data set.

As shown in Figure 7, the model proposed in this paper has the best performance in all the comparison models, followed by reference [21], and the worst performance is reference [15]. When the number of iterations of the proposed detection model reaches 500, the accuracy has reached 97.0%, while the accuracies of reference [21] and reference [15] are still less than 96.0% after 700 iterations. This shows that it is correct to improve the embedding layer in this paper. In addition, sentence segmentation embedding layer and position embedding layer are added to optimize and analyze the sample data set, so as to provide effective and reliable data input support for Bi-LSTM-CNN model and ensure that the proposed model can effectively identify rumors.

Further, the loss value of each model is analyzed, as shown in Figure 8.

It can be seen in Figure 8 that the convergence of the proposed detection model is stronger than that of the

Table 2: Sensitivity analysis of convolution kernel parameters.

(a) Convolution kernel size

Size	Average F_1
(1,2,4)	0.8532
(2,4,6)	0.8653
(4,6,8)	0.8775
(6,8,10)	0.8542

(b) Number of convolution kernels

Quantity	Average F_1
64	0.8234
128	0.8456
256	0.8511
512	0.8432

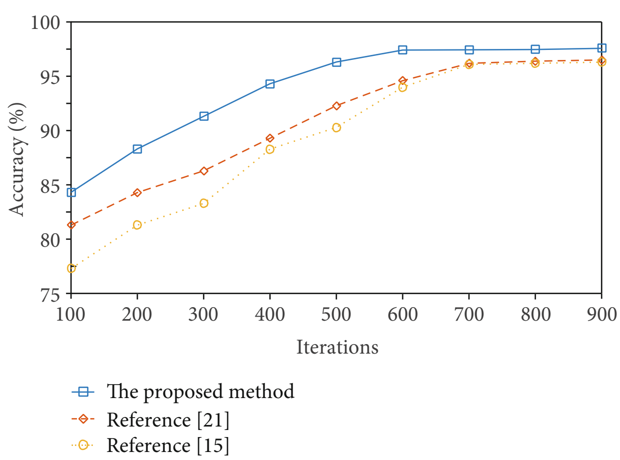

Figure 7: Rumor detection accuracy of different methods.

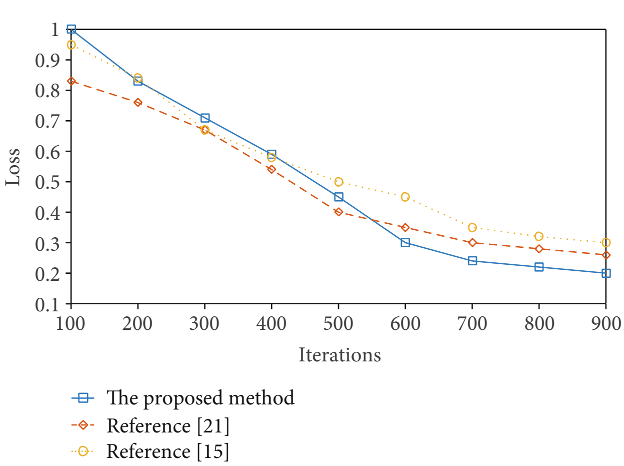

Figure 8: Loss value under different methods.

Table 3: Performance analysis of different methods.

Method	Precision Pre	Recall Re	F_1	Accuracy Acc
Proposed method	0.9771	0.8563	0.8723	0.9726
Reference [21]	0.9632	0.8342	0.8642	0.9611
Reference [15]	0.9523	0.8411	0.8471	0.9601

comparative methods, making the classification faster. Bi-LSTM-CNN model combines Bi-LSTM with CNN to extract text features and adds the influence of subsequent sequence information to the learning task. With in-depth learning, the wider the data features obtained, the more accurate the text representation, and the better the classification effect. Therefore, it is confirmed that the improvement of embedded layer and the model fusion of Bi-LSTM-CNN can realize the accurate and efficient identification of network rumors.

At the same time, this paper also analyzes each model by synthesizing each evaluation index, as shown in Table 3.

As shown in Table 3, compared with the current rumor detection methods, the network rumor detection method based on deep learning network proposed in this paper has better performance. The precision Pre of the method proposed in this paper is 0.9771, the recall Re is 0.8563, the F_1 is 0.8723, and the accuracy Acc is 0.9726. All indexes are higher than those in the comparative methods. This is because the proposed method integrates Bi-LSTM and CNN, realizes the effective screening of data features of rumor sample set, has strong feature extraction ability for sequence data, and ensures high precision and high accuracy.

5. Conclusion

In order to improve the reliability of social media information, this paper proposes a network rumor detection and judgment method based on deep learning network. When the proposed detection method retains the original word embedding layer in the embedding layer, the sentence segmentation embedding layer and position embedding layer are introduced, and the multilayer embedding hierarchical processing analysis is realized for the original sample data set, so as to provide reliable and complete data support for the subsequent rumor detection model. The experimental results show that the proposed method can effectively detect network rumors and is of great significance to the construction of green network.

With the advent of 5G era, it is easy for people to choose to use video and pictures to spread information. According to the statistics of microblog platform, the microblog messages with pictures or video information are eleven times more than those only with texts. The vectorization representation of video and picture is very different from that of text. How to efficiently detect video information and picture content and realize multimodal rumor detection has become the focus and difficulty in the current research and will also become the direction and content of the next research in this paper.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The author declares that there is no conflict of interest regarding the publication of this paper.

References

[1] X. Y. Chen, D. D. Zhu, D. Z. Lin, and D. Cao, "Rumor knowledge embedding based data augmentation for imbalanced rumor detection," *Information Sciences*, vol. 580, no. 1, pp. 352–370, 2021.

[2] T. T. Nguyen, T. T. Nguyen, T. T. Nguyen, B. Vo, J. Jo, and Q. V. H. Nguyen, "JUDO: just-in-time rumour detection in streaming social platforms," *Information Sciences*, vol. 570, no. 6380, pp. 70–93, 2021.

[3] Y. H. Chen, C. Y. Yin, and W. L. Zuo, "Multi-task learning for stance and early rumor detection," *Optical Memory and Neural Networks*, vol. 30, no. 2, pp. 131–139, 2021.

[4] W. Wang, Y. Qiu, S. Xuan, and W. Yang, "Early rumor detection based on deep recurrent Q-learning," *Security and Communication Networks*, vol. 2021, no. 2, Article ID 5569064, 13 pages, 2021.

[5] M. C. Guo, Z. W. Xu, L. M. Liu, M. Guo, and Y. Zhang, "An adaptive deep transfer learning model for rumor detection without sufficient identified rumors," *Mathematical Problems in Engineering*, vol. 2020, no. 1, Article ID 7562567, 12 pages, 2020.

[6] K. F. Tu, C. Chen, C. Y. Hou, J. Yuan, J. Li, and X. Yuan, "Rumor2vec: a rumor detection framework with joint text and propagation structure representation learning," *Information Sciences*, vol. 560, no. 1, pp. 137–151, 2021.

[7] F. Xu, V. S. Sheng, and M. W. Wang, "Near real-time topic-driven rumor detection in source microblogs," *Knowledge-Based Systems*, vol. 207, no. 6380, pp. 106391–106399, 2020.

[8] D. Z. Lin, B. Ma, D. L. Cao, and S. Li, "Chinese microblog rumor detection based on deep sequence context," *Concurrency and Computation Practice and Experience*, vol. 31, no. 23, pp. 1–12, 2019.

[9] C. H. Song, C. Yang, H. M. Chen, C. Tu, Z. Liu, and M. Sun, "CED: credible early detection of social media rumors," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 8, pp. 3035–3047, 2021.

[10] S. A. Alkhodair, S. H. H. Ding, B. C. M. Fung, and J. Liu, "Detecting breaking news rumors of emerging topics in social media," *Information Processing & Management*, vol. 57, no. 2, p. 102018, 2020.

[11] Z. Y. Wu, D. C. Pi, J. F. Chen, M. Xie, and J. Cao, "Rumor detection based on propagation graph neural network with attention mechanism," *Expert Systems with Applications*, vol. 158, p. 113595, 2020.

[12] X. F. Dong, Y. Lian, Y. X. Chi, X. Tang, and Y. Liu, "A two-step rumor detection model based on the supernetwork theory about Weibo," *The Journal of Supercomputing*, vol. 77, no. 10, pp. 12050–12074, 2021.

[13] A. E. Fard, M. Mohammadi, Y. Chen et al., "Computational rumor detection without non-rumor: a one-class classification approach," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 5, pp. 1–17, 2019.

[14] H. K. Thakur, A. Gupta, A. Bhardwaj, and D. Verma, "Rumor detection on Twitter using a supervised machine learning framework," *International Journal of Information Retrieval Research*, vol. 8, no. 3, pp. 1–13, 2018.

[15] N. Kandasamy and K. Murugasamy, "Rumor detection on Twitter using a supervised machine learning framework," *Concurrency and Computation-Practice & Experience*, vol. 33, no. 20, pp. 1–10, 2021.

[16] S. Srinivasan and B. Dhinesh, "A parallel neural network approach for faster rumor identification in online social networks," *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 15, no. 4, pp. 69–89, 2019.

[17] J. Choi, T. Ko, Y. Choi, H. Byun, and C. K. Kim, "Dynamic graph convolutional networks with attention mechanism for rumor detection on social media," *PLoS One*, vol. 16, no. 8, pp. 1–17, 2021.

[18] T. H. Ma, H. H. Zhou, Y. Tian, and N. al-Nabhan, "A novel rumor detection algorithm based on entity recognition, sentence reconfiguration, and ordinary differential equation network," *Neurocomputing*, vol. 447, no. 1, pp. 224–234, 2021.

[19] S. Santhoshkumar and L. D. D. Babu, "Earlier detection of rumors in online social networks using certainty-factor-based convolutional neural networks," *Social Network Analysis and Mining*, vol. 10, no. 1, pp. 1–17, 2020.

[20] X. Y. Chen, L. Ke, Z. P. Lu, H. Su, and H. Wang, "A novel hybrid model for Cantonese rumor detection on Twitter," *Applied Sciences*, vol. 10, no. 20, p. 7093, 2020.

[21] N. Agarwal, M. A. Wani, and P. Bours, "Impact of unreliable content on social media users during COVID-19 and stance detection system," *Electronics*, vol. 10, no. 1, pp. 1–21, 2020.

[22] L. J. Lin and Z. Y. Chen, "Social rumor detection based on multilayer transformer encoding blocks," *Concurrency and Computation-Practice & Experience*, vol. 33, no. 6, pp. 1–15, 2020.

[23] W. Ai, K. L. Li, and K. Q. Li, "An effective hot topic detection method for microblog on spark," *Applied Soft Computing*, vol. 70, no. 1, pp. 1010–1023, 2018.

[24] J. Z. Wang, Y. Zhou, W. Zhang, R. Evans, and C. Zhu, "Concerns expressed by Chinese social media users during the COVID-19 pandemic: content analysis of Sina Weibo microblogging data," *Journal of Medical Internet Research*, vol. 22, no. 11, pp. 1–13, 2020.

[25] J. Ma, W. Gao, S. Joty, and K. F. Wong, "An attention-based rumor detection model with tree-structured recursive neural networks," *ACM Transactions on Intelligent Systems and Technology*, vol. 11, no. 4, pp. 1–28, 2020.

[26] Z. H. Wang and Y. Guo, "Empower rumor events detection from Chinese microblogs with multi-type individual information," *Knowledge and Information Systems*, vol. 62, no. 9, pp. 3585–3614, 2020.

[27] Y. C. Xu, C. Wang, Z. P. Dan et al., "Deep recurrent neural network and data filtering for rumor detection on Sina Weibo," *Symmetry*, vol. 11, no. 11, pp. 1–11, 2019.