International Journal of Data Warehousing and Mining
Volume 19 • Issue 2

Research on Rumor Detection Based on a Graph Attention Network With Temporal Features

Xiaohui Yang, Hebei University, China
Hailong Ma, Hebei University, China & China Telecom Stocks Co., Ltd., China*
https://orcid.org/0000-0002-6345-4005
Miao Wang, Hebei University, China

ABSTRACT

The higher-order and temporal characteristics of tweet sequences are often ignored in the field of rumor detection. In this paper, a new rumor detection method (T-BiGAT) is proposed to capture the temporal features between tweets by combining a graph attention network (GAT) and gated recurrent neural network (GRU). First, timestamps are calculated for each tweet within the same event. On the premise of the same timestamp, two different propagation subgraphs are constructed according to the response relationship between tweets. Then, GRU is used to capture intralayer dependencies between sibling nodes in the subtree; global features of each subtree are extracted using an improved GAT. Furthermore, GRU is reused to capture the temporal dependencies of individual subgraphs at different timestamps. Finally, weights are assigned to the global feature vectors of different timestamp subtrees for aggregation, and a mapping function is used to classify the aggregated vectors.

KEYWORDS

Gated Recurrent Neural Network, Graph Attention Network, Rumor Detection, Temporal Features, Timestamp

INTRODUCTION

From the 20th century to the present, the world industrial pattern has gradually tilted toward Internet-related fields. Many IT companies, such as Microsoft, Google, and Alibaba, began to rise rapidly. They do not hesitate to invest huge sums of money and recruit a large number of researchers to seize new fields. At present, the information dissemination carrier represented by Twitter has become the main tool for people to communicate. Users can communicate through social software without leaving home and learn about major events in the world. However, people with ulterior motives have begun to spread rumors with the help of social networks, making it difficult for users to distinguish between true and rumor without their knowledge. Since rumors cover a very wide range and users

DOI: 10.4018/IJDWM.319342 *Corresponding Author

who publish rumors are very concealed, it is very difficult to supervise them. At present, Baidu, Tencent, Weibo, and other well-known Internet companies have established rumor-refuting platforms. Various Internet platforms organize researchers to explore efficient rumor detection methods that can adapt to the big data environment. Text features (Azri et al., 2021; GuangJun et al., 2020; Li et al., 2022; Ma et al., 2022; Shelke & Attar, 2022; Xu et al., 2021), image features (Azri et al., 2021; Li et al., 2022), user features (Shelke & Attar, 2022), and spread features (Ma et al., 2022) have become mainstream research directions.

To adapt to the environmental requirements of big data, the related methods of rumor detection are gradually transferred from manual-based related methods to machine learning-based related methods. Since the related methods based on machine learning cannot model the social relations of users, this method cannot effectively extract the high-level and abstract features of rumors. In 2016, Kipf & Welling (2016) proposed graph convolutional neural networks. The related methods of graph neural networks gradually entered the field of view of many scholars and achieved good performance. Since the graph convolutional neural networks(GCN) needs to introduce an adjacency matrix, out-degree nodes and in-degree nodes need to participate in the node aggregation process at the same time, which limits the aggregation direction. At this stage, the related methods of rumor detection mainly consist of related methods based on machine learning and related methods based on graph neural networks.

Related technologies based on machine learning have been very mature. Most scholars use classifiers or classification functions to determine whether tweets are rumors by extracting relevant features and inputting them into trained models. Ma et al. (2016) captured the time-varying contextual features through a recurrent neural network and proposed a rumor detection model that fuses temporal feature information; Shi et al. (2018) not only improved the detection efficiency but also solved the problem of data sparseness by fusing the recurrent neural network with the topic features of emergencies; Min et al. (2016) combined a momentum model and temporal analysis-based method to filter fake microblogs. These methods demonstrate the importance of temporal features in rumor detection by extracting temporal relationships between tweets or keywords. Gao et al. (2020) used task-specific features based on bidirectional language models to learn contextual embedded textual information and event sequence information; Liu et al. (2020) used deep learning to extract the text features of tweets, image features and text information in images. However, these kinds of methods only stay at the most basic surface features and cannot extract high-level, abstract global features of rumors.

The related techniques of machine learning have achieved great success in dealing with the problem of Euclidean space, but the results of processing in non-Euclidean space are unsatisfactory. Related graph neural network methods have been developed, which can solve these problems very well. Graph neural networks usually model information as a graph structure and then extract relevant features. This idea can effectively extract the dependencies that exist between different individuals. Xue et al. (2021) extracted user features, propagation features, and text features by combining GAT and multimodal gating unit and proposed a multifeature rumor detection model; Lotfi et al. (2021) constructed a user graph and a tweet graph, captured user line features and tweet response relationship features through a graph convolutional neural network, and concatenated the two feature vectors and used a classifier to discriminate. Related methods based on graph neural networks extract high-order and abstract features by converting tweets or words into individual nodes and through node aggregation. Different from traditional deep learning methods, they lack the temporal expression between tweets.

Among the existing research methods of rumors, most people have begun to study temporal features. As the graph neural network has only ushered in a development climax in recent years, the research methods of centralized temporal features are mainly distributed in the traditional deep learning method, and there are few studies on the temporal features of sibling nodes. Typically, within the same hierarchy, tweets posted first have some influence on tweets posted later. There is no perfect solution for fully expressing the relationship between sibling nodes and effectively extracting the temporal features of the event life cycle.

This paper proposes a rumor detection model T-BiGAT based on GAT with temporal features for social networks, which can effectively capture the temporal features between tweets. Experiments show that T-BiGAT is better than the baseline method in terms of evaluation indicators.

The main contributions of this paper are as follows:

1. The temporal between tweets is comprehensively considered, and the temporal dependencies are extracted by constructing a subtree of timestamps and introducing the gated recurrent neural network (GRU) model twice.
2. Integrating user credibility into the feature extraction process of graph attention networks. This method can calculate the credibility score according to the relevant features of the user and adaptively assign the weight according to the user credibility and the node weight.
3. A new rumor detection model (T-BiGAT) is proposed. Relevant experiments show that T-BiGAT outperforms the baseline method in both the overall performance comparison analysis and the ablation experiments in the real datasets.

T-BIGAT RUMOR DETECTION MODEL

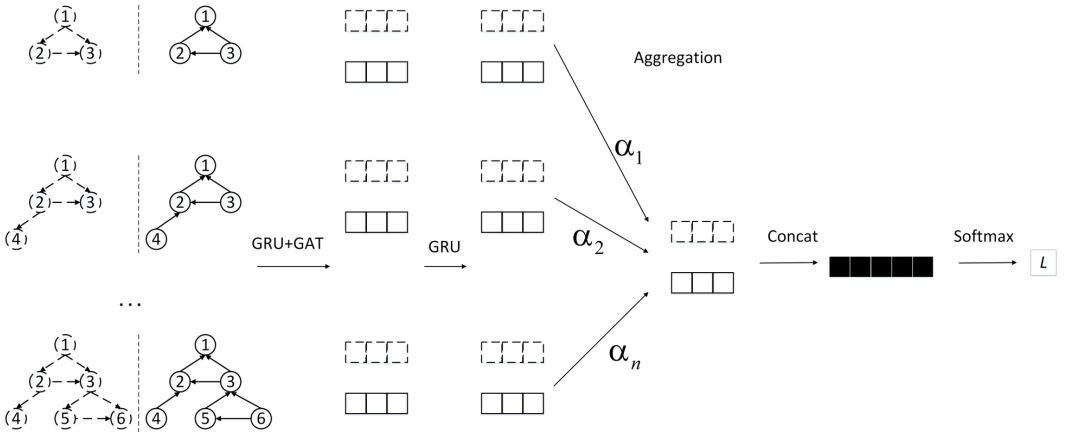
A rumor detection model that explores the temporal relationship between tweets is proposed, as shown in Figure 1. First, the propagation subgraphs and diffusion subgraphs under different timestamps are constructed, and the GRU model is used to capture the temporal features between sibling nodes. Then, the improved GAT model is used to capture the global features of the propagation map and diffusion map under different timestamps, and the GRU model is used again to capture the inherent temporal features of temporal information with different timestamps. Finally, based on the attention mechanism, weights are assigned to the feature vectors of different subgraphs for feature aggregation, and the mapping function is used to determine whether the event is a rumor. The model proposed in this paper is introduced in turn from feature extraction, rumor detection classification, and experiments.

Feature Extractions

Construction of the Graph

It is difficult to verify whether the event is a rumor from just the first few tweets. Source tweets often generate a flood of response tweets over time. To better represent the temporal features, this paper

Figure 1.
Structure Diagram of the T-BiGAT Rumor Detection Model



divides the source tweets and corresponding response tweets into multiple parts according to timestamps. Each section corresponds to a subgraph at a different timestamp. Suppose $timeFirst_i$ and $timeLast_i$ are the earliest published source tweet time and the latest published response tweet time, respectively. Then, this model converts the time when a tweet was posted into a timestamp index value between 0 and N. The specific formula is as follows:

$$Interval = \frac{timeLast_i - timeFirst_i}{N} \tag{1}$$

$$TS\left(m_{ij}\right) = \frac{t_{m_{ij}} - timeFirst_i}{Interval} \tag{2}$$

where Interval represents the length of the time interval; N is an adjustable variable of the time interval;TS() represents the index value of the timestamp of tweet, $t_{m_{ij}}$ represents the time when tweet was published.

In the process of constructing a subgraph, there is usually a subtree consisting of a single node or several nodes at a certain timestamp, resulting in strong tree sparsity and insufficient feature extraction. Therefore, this model connects the nodes under each timestamp as a subtree. The specific method is to take the source tweet as the root node under each timestamp and the response tweet under the timestamp as the child node. If there is a lack of intermediate nodes between the child node and the root node, the intermediate nodes under other timestamps are stored in the subtree to realize that there is a directed path between the child node and the root node under the timestamp.

Each tweet is encoded using the bert model published by Google (Devlin et al., 2018). The feature vector of each tweet acts as a vector in each node. The response relationship between tweets acts as an edge in the tree. Within the same level of the tree, tweets posted earlier usually have an impact on tweets posted later. Therefore, when judging the intralayer dependency between sibling nodes, the cosine similarity is used to calculate the direct correlation of nodes. If the cosine similarity score is greater than T, then it is considered that there is an intralayer dependency between sibling nodes. Therefore, a directed connection is made to the sibling nodes in the tree; otherwise, the relationship is ignored. Among them, the time sequence of tweet generation is the direction of the edge. It is assumed that the nodes are labeled in the order in which they are generated in time. Build the rumor diffusion graph TD-Graph shown in the solid line part and the rumor propagation graph BU-Graph in the dotted line part in Figure 2.

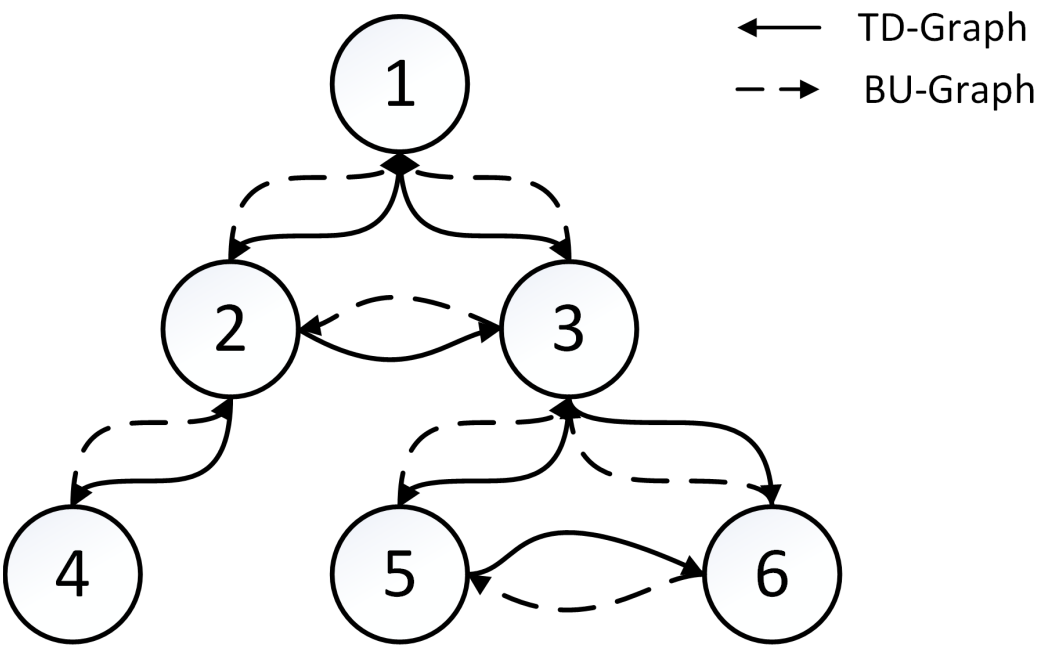
Feature Extraction of Rumors

The features of rumors mainly include propagation features and diffusion features. This model performs feature extraction on the constructed rumor structure graph based on GAT. First, after constructing multiple subgraphs based on different timestamps, neighbor nodes with temporal relationships in the subgraphs are sequentially input into the GRU model, and the neighbor nodes are updated to obtain temporal intralayer dependencies. Then, the user credibility is integrated into the weight of the feature extraction of the GAT model, and the global features under different timestamps are extracted. Finally, the GRU model is introduced to capture the relationship of continuous timestamp subtrees and use the attention mechanism for aggregation. The specific process is as follows.

Suppose the update process of the t-th node of the propagation subtree with the N-th timestamp has n nodes as an example. The input layer is:

$$\left(S_{1(N)}^{(t-1)}, S_{2(N)}^{(t-1)}, ..., S_{n(N)}^{(t-1)}\right), S_{i(N)}^{(t-1)} \in R^F \tag{3}$$

Figure 2.
Rumor Tree Structure Diagram



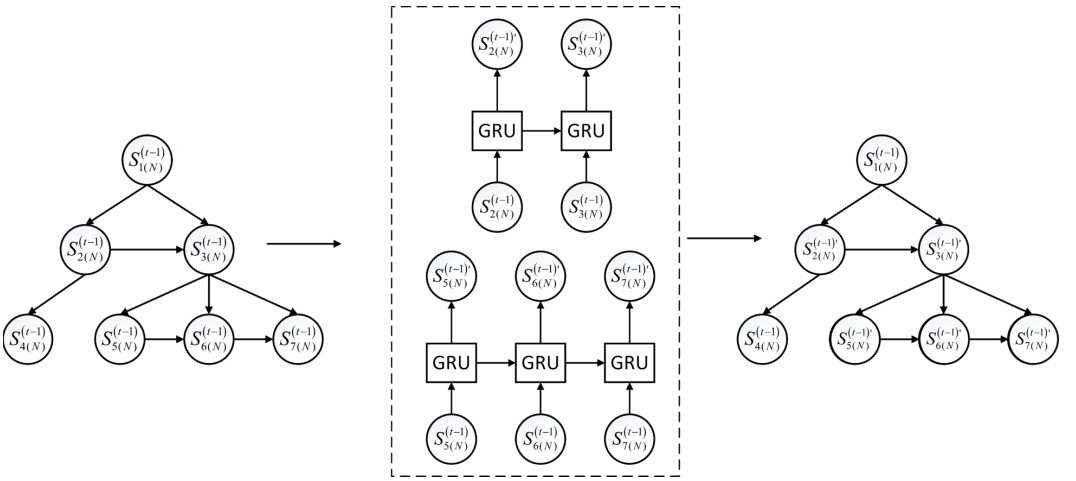
Tweets posted first often have an impact on tweets posted later. In the construction of the tree, if there are edges with temporal relationships between sibling nodes, the GRU (Cho et al., 2014) is introduced to capture the temporal relationship among sibling nodes. The neighbor nodes with time-series dependencies are sorted chronologically and input to the GRU model in turn to update the neighbor nodes. It is assumed that the update process of the sibling nodes of the timestamp subtree with a temporal relationship is shown in Figure 3.

It is assumed that $\left(S_{2(N)}^{(t-1)}, S_{3(N)}^{(t-1)}\right)$ and $\left(S_{5(N)}^{(t-1)}, S_{6(N)}^{(t-1)}, S_{7(N)}^{(t-1)}\right)$ are both neighbor nodes with time-series dependencies in chronological order. The neighbor nodes sorted according to the time sequence are input to the GRU model in turn, and the neighbor nodes are updated to obtain the representation $\left(S_{2(N)}^{(t-1)'}, S_{3(N)}^{(t-1)'}\right)$ and $\left(\mathrm{S}_{5(\mathrm{N})}^{(\mathrm{t}-1)'}, \mathrm{S}_{6(\mathrm{N})}^{(\mathrm{t}-1)'}, \mathrm{S}_{7(\mathrm{N})}^{(\mathrm{t}-1)'}\right)$ of the neighbor nodes with the dependencies in the temporal layer. For the sake of the same symbol and convenience for the following representation, $S_{i(N)}^{(t-1)}$ and $S_{i(N)}^{(t-1)'}$ are always represented by $h_{i(N)}^{(t-1)}$; then, the nodes of the output layer are represented as:

$$\left(h_{1(N)}^{t}, h_{2(N)}^{t}, \ldots, h_{n(N)}^{t}\right), h_{i(N)}^{t} \in R^{F'} \tag{4}$$

Usually, the importance of neighbor nodes is closely related to user credibility. A credible user's response tweet should have a higher weight on the event, and a rumor-spreading user's response tweet should have a lower weight on the event. Based on this idea, this model comprehensively considers the user credibility and text features of neighbor nodes when performing feature aggregation and integrates user credibility into the node aggregation process to represent the importance of neighbor nodes. The user credibility calculation formula is as follows:

Figure 3.
Temporal Sibling Node Update Process



$$f_{influence}\left(u_j\right)=\frac{C_{biflowers}\left(u_j\right)}{C_{flowers}\left(u_j\right)} \tag{5}$$

$$f_{verified}\left(u_j\right)=\begin{cases}0 & unverified\\ 1 & verified\end{cases} \tag{6}$$

$$f_{InfoIntegrity}\left(u_j\right)=\begin{cases}0 & incomplete\\ 1 & complete\end{cases} \tag{7}$$

$$f_{credibility}\left(u_j\right)=f_{\text{inf luence}}\left(u_j\right)+f_{verified}\left(u_j\right)+f_{InfoIntegrity}\left(u_j\right) \tag{8}$$

$f_{influence}\left(u_j\right)$ represents the influence score of user u_j; $C_{biflowers}\left(u_j\right)$ represents the number of people who follow each other by user u_j; $C_{flowers}\left(u_j\right)$ represents the number of people who follow each other by user u_j; $f_{verified}\left(u_j\right)$ represents the verification score of user u_j; if the user has been officially verified, then the value is 1; otherwise, the value is 0; $f_{InfoIntegrity}\left(u_j\right)$ represents the information integrity score of user u_j; if the user information is complete, the value is 1; otherwise, the value is 0. After obtaining the above three formula scores, the user's credibility $f_{credibility}\left(u_j\right)$ is obtained by adding.

To accurately judge the importance of neighbor nodes, this model comprehensively considers the relationship between the tweet feature vector and user credibility and integrates the user credibility score $f_{credibility}\left(u_j\right)$ into the weight of the GAT model to participate in node updating. To prevent the feature vector updated by the node from being too large or too small, the user credibility score is normalized. The specific formula is as follows:

$$\beta_j=\frac{f_{credibility}\left(u_j\right)}{\sum_{j=1}f_{credibility}\left(u_j\right)} \tag{9}$$

The GAT model node update formula is as follows:

$$h_{i(N)}^{t} = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in N_i} W_{(N)}^{k}\alpha_{ij(N)}h_{j(N)}^{(t-1)}\right) \tag{10}$$

N_i is all in-degree nodes connected to node i in the subtree; $h_{i(N)}^{t}$ represents the t-th update result of node i at the n-th timestamp; K means that the multihead attention mechanism includes a total of K heads; $W_{(N)}^{k}$ represents the trainable weight matrix of the node under the Kth feature vector at the Nth timestamp; $\alpha_{ij(N)}$ is the matrix forwarded by all nodes in the subtree at the Nth timestamp.

The user credibility β_j is incorporated into the weight of the node update formula of the GAT model to measure the importance of neighbor nodes. The new node update formula in the GAT model is:

$$h_{i(N)}^{t} = \sigma\left\{\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in N_i} W_{(N)}^{k}\left[\gamma\alpha_{ij(N)} + \left(1-\gamma\right)\beta_j\right]h_{j(N)}^{(t-1)}\right\} \tag{11}$$

γ is the weight coefficient to measure between $\alpha_{ij(N)}$ and β_j. Source tweets always have deeper influence than response tweets, and source tweets can better represent the original information content, so this model adds root node feature enhancement. The specific method is that the output feature of each node is added to the root node feature of the previous moment. The specific formula is as follows:

$$h_{i(N)}^{t} = \sigma\left\{\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in N_i} W_{(N)}^{k}\left[\gamma\alpha_{ij(N)} + \left(1-\gamma\right)\beta_j\right]h_{j(N)}^{(t-1)}\right\} + h_{root(N)}^{(t-1)} \tag{12}$$

Based on the above feature extraction method, this model models the feature vector representations $\left(h_{1(N)}^{BU}, h_{2(N)}^{BU}, ..., h_{n(N)}^{BU}\right)$ and $\left(h_{1(N)}^{TD}, h_{2(N)}^{TD}, ..., h_{3(N)}^{TD}\right)$ of each node of the diffusion subgraph and the propagation subgraph at the Nth timestamp. The feature vectors of nodes in the diffusion graph and the propagation graph are aggregated by using an average pooling operation. The formula is expressed as follows:

$$G_N^{BU} = MEAN\left(h_{1(N)}^{BU}, h_{2(N)}^{BU}, ..., h_{n(N)}^{BU}\right) \tag{13}$$

$$G_N^{TD} = MEAN\left(h_{1(N)}^{TD}, h_{2(N)}^{TD}, ..., h_{3(N)}^{TD}\right) \tag{14}$$

G_N^{BU} is the propagation feature of the subgraph at the nth timestamp obtained based on GRU and GAT; G_N^{TD} is the diffusion feature of the subgraph at the nth timestamp obtained based on GRU and GAT. Based on the above steps, the model obtains the temporal information $\left(G_1^{BU}, G_2^{BU}, ..., G_N^{BU}\right)$ and $\left(G_1^{TD}, G_2^{TD}, ..., G_N^{TD}\right)$ based on the propagation features and diffusion features of different timestamps.

Since the reset gate and update gate in the GRU model can effectively discard the unimportant information at the last moment, the important information at the last moment is selected to be fused with the input information at this moment to improve the temporal feature correlation between the

sequence information. In this paper, the GRU model is introduced again, and the temporal information of the propagation feature and the diffusion feature are sequentially input into the GRU model to capture the correlation in the temporal information. The formula is expressed as follows:

$$\left(H_1^{BU}, H_2^{BU}, ..., H_N^{BU}\right) = GRU\left(G_1^{BU}, G_2^{BU}, ..., G_N^{BU}\right) \tag{15}$$

$$\left(H_1^{TD}, H_2^{TD}, ..., H_N^{TD}\right) = GRU\left(G_1^{TD}, G_2^{TD}, ..., G_N^{TD}\right) \tag{16}$$

Finally, the attention mechanism acts as a channel for the aggregation of propagating features and diffusing features of consecutive timestamps. The formula is expressed as follows:

$$H^{TD} = \sum_{n=1}^{N} \alpha_n H_N^{TD} \tag{17}$$

$$H^{BU} = \sum_{n=1}^{N} \alpha_n H_N^{BU} \tag{18}$$

H^{TD} and H^{BU} are the propagation features and diffusion features extracted by node update through the improved graph attention network, respectively.

Feature Aggregation and Classification

Global features H^{TD} and H^{BU} of different directions for each event of the fused temporal features are concatenated:

$$H = Concat\left(H^{TD}, H^{BU}\right) \tag{19}$$

The feature vector H is connected to the fully connected layer, and the final output layer mapping function adopts the softmax function. The calculation formula is as follows:

$$Y = Softmax\left(W_l H + b_l\right) \tag{20}$$

where Y represents the category of the predicted source tweet in each dataset, W_l is the trainable matrix, and b_l is the intercept term of the softmax function. During training, all model parameters use an L2 loss function.

EXPERIMENTS

In this subsection, the authors introduce the datasets and performance evaluation indicators used in this experiment. The detection performance of this model was compared with several baseline models on the public dataset, and various factors that affected the final detection result were subjected to ablation experiments. In addition, at the beginning of the event, a deadline was set to explore whether T-BiGAT can efficiently and quickly detect rumors.

Dataset Source and Processing

Since this method involves user reliability and temporal features, the Weibo dataset (Ma et al., 2016) and Pheme dataset (Zubiaga et al., 2016) were used as public datasets for this comparative experiment. The Weibo dataset includes a total of 2351 nonrumors and 2313 rumors. The Pheme dataset contains

five emergencies: Charlie Hedbo, Germanwings Crash, Ottawa Shooting, Sydney Siege, and Ferguson. Each event has a separate directory. There are two folders in each directory, rumored or nonrumored source tweets. Each source tweet in turn has multiple unflagged response tweets. Each tweet is saved in.jason format. The specific dataset statistics are shown in Tables 1 and 2.

The Weibo dataset and Pheme dataset contain considerable useless information. In the process of data preprocessing, this model deletes all hyperlinks, pictures, attribute information that is not involved in this model, and reference information in response tweets contained in the dataset through regular expressions and performs noise reduction processing on the dataset. Accuracy, precision, recall, and F1 value are selected as model evaluation indicators. The calculation formula is as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{21}$$

$$Precision = \frac{TP}{TP + FP} \tag{22}$$

$$Recall = \frac{TP}{TP + FN} \tag{23}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{24}$$

The specific meanings of TP, TN, FP, and FN are shown in Table 3.

Parameter Settings

The experimental settings were as follows: the functions of this model were implemented in Python language, and the experimental operating environment was Linux operating system, IntelI CITM) i7-85500H CPU @ 2.3GHz, and 16 GB memory.

Table 1.
Pheme Dataset Statistics

Event	Source tweet rumor count	Source tweet nonrumor count
Charlie Hedbo	458	1621
Ferguson	284	859
Germanwings Crash	238	231
Ottawa Shooting	470	420
Sydney Siege	522	699
Total	1972	3830

Table 2.
Weibo Dataset Statistics

	Number
total number of events	4664
total number of users	2746818
Total number of tweets	3805656
number of real information	2351
Number of false rumors	2313

Table 3.
Symbol Specific Representation

Predicted Value\Actual Value	1	0
1	TP	FP
0	FN	TN

The T-BiGAT model was set according to the relevant parameters of the best performance on the training set, and the number of neural network iterations was set to 300, that is, epoch=300. The data volume of each iteration was batch-size=256, the learning rate was $\eta = 0.001$, droupout=0.2, and the cosine similarity T=0.8. A two-layer GAT model was used, and each layer included K=4 attention heads. Additionally, the Adam optimizer was added during training.

Overall Performance Analysis

The baseline methods DTC, SVM-TS, RvNN, BiGCN, and GRU-RVNN were selected for comparison with the T-BiGAT rumor detection model to highlight the performance of the T-BiGAT model. The results are shown in Table 4 and Table 5.

- **GRU-RNN (Ma et al., 2016):** A rumor detection model that learns correlations between posts through a gated recurrent neural network.
- **BiGCN (Bian et al., 2020):** This method performs feature extraction in different directions in the constructed undirected tree through a graph convolutional neural network (GCN).
- **DTC (Castillo et al., 2011):** The method is based on the decision tree method of machine learning to judge the constructed text feature vector.
- **SVM-TS (Ma et al., 2015):** This method introduces the temporal features of events by constructing a timestamp structure and uses the machine learning-based SVM classifier for judgment.
- **RvNN (Ma et al., 2018):** This method converts all tweets in an event into tree structures with two different directions, and each node is a vector representation of a tweet in the event. The signals of different branches in the tree are captured by a recurrent neural network.

Table 4.
Comparison of Rumor Detection Performance in the Pheme Dataset

Method	Label	Acc	Prec	Recall	F1
DTC	T F	0.582	0.582 0.579	0.573 0.588	0.578 0.584
SVM-TS	T F	0.651	0.663 0.642	0.617 0.686	0.639 0.663
RvNN	T F	0.820	0.722 0.869	0.741 0.857	0.731 0.867
GRU-RNN	T F	0.775	0.667 0.825	0.643 0.840	0.658 0.832
BiGCN	T F	0.835	0.851 0.811	0.798 0.862	0.830 0.837
T-BiGAT	T F	0.882	0.898 0.873	0.919 0.864	0.887 0.878

Table 5.
Comparison of Rumor Detection Performance in the Weibo Dataset

Method	Label	Acc	Prec	Recall	F1
DTC	T F	0.831	0.815 0.847	0.847 0.815	0.830 0.831
SVM-TS	T F	0.857	0.878 0.839	0.830 0.885	0.857 0.861
RvNN	T F	0.910	0.952 0.876	0.864 0.956	0.906 0.914
GRU-RNN	T F	0.775	0.825 0.667	0.840 0.643	0.832 0.659
BiGCN	T F	0.960	0.961 0.959	0.960 0.963	0.959 0.960
T-BiGAT	T F	0.972	0.962 0.983	0.976 0.969	0.980 0.964

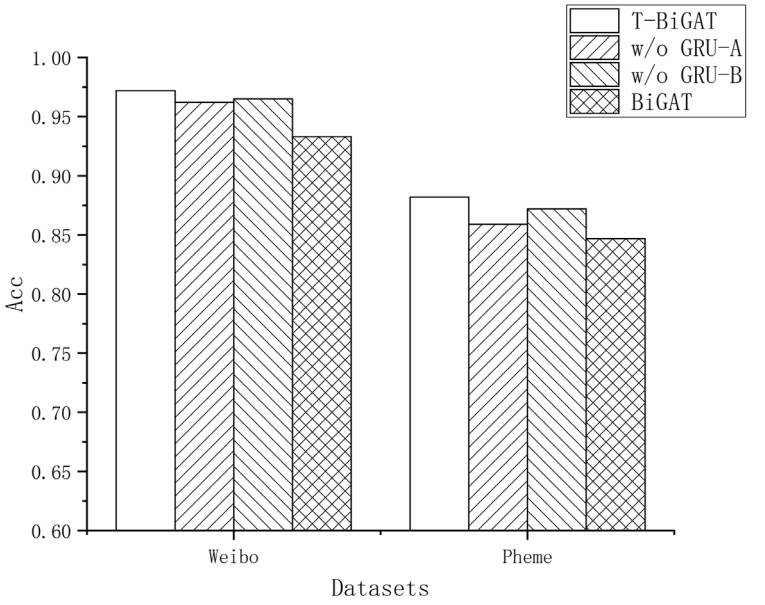
The T-BiGAT model outperformed the comparative baseline methods on both the Weibo dataset and the Pheme dataset. Among them, DTC and SVM-TS based on manual feature extraction methods performed the worst on two public datasets. The method of manual feature extraction is to design features based on human thoughts rather than the idea of computer training and learning. This method requires a lot of manpower to label the dataset, so that the final detection result has a great relationship with the accuracy of the original dataset labeling. RvNN and GRU-RNN based on machine learning methods were able to adapt to the big data environment and overcome the shortcomings of rumor detection models based on manual methods. Therefore, the RvNN and GRU-RNN rumor detection models are better than the DTC and SVM-TS rumor detection models. BiGCN based on the graph neural network method has obvious advantages over all baseline methods. This method aggregates response tweets and source tweets in two different directions through the component graph method, making BiGCN the best detection effect among all baseline methods. However, the T-BiGAT rumor detection model proposed in this chapter outperformed all the compared baseline methods. Compared with other methods, this model builds multiple directed graphs based on multiple timestamps and introduces the GRU model to extract temporal features. The method not only considers the temporal dependencies between sibling nodes but also considers the temporal correlations between global features extracted based on subgraphs under different timestamps. In addition, this model also introduces user credibility into the feature extraction process of the graph attention network to select credible neighbor nodes, resulting in the T-BiGAT model outperforming the comparative baseline methods on the two datasets. In summary, the effectiveness of all the methods in this paper can be proven.

Ablation Experiment

Aiming at whether it is necessary for the T-BiGAT rumor detection model to use the GRU model twice to introduce temporal features, this experiment compared the T-BiGAT model with the variant models BiGAT, w/o GRU-A, and w/o GRU-B, judged by the accuracy:

- **w/o GRU-A:** This variant model considers temporal features between sibling nodes in the same subtree but does not consider temporal features between subtrees with different timestamps.
- **w/o GRU-B:** This variant model considers temporal features between different timestamp subtrees but does not consider temporal features between sibling nodes in the same subtree.
- **BiGAT:** This variant model neither considers the temporal features between sibling nodes in the same timestamp subtree nor between different timestamp subtrees.

Figure 4.
Influence of the Introduction of the GRU Model on Detection Results

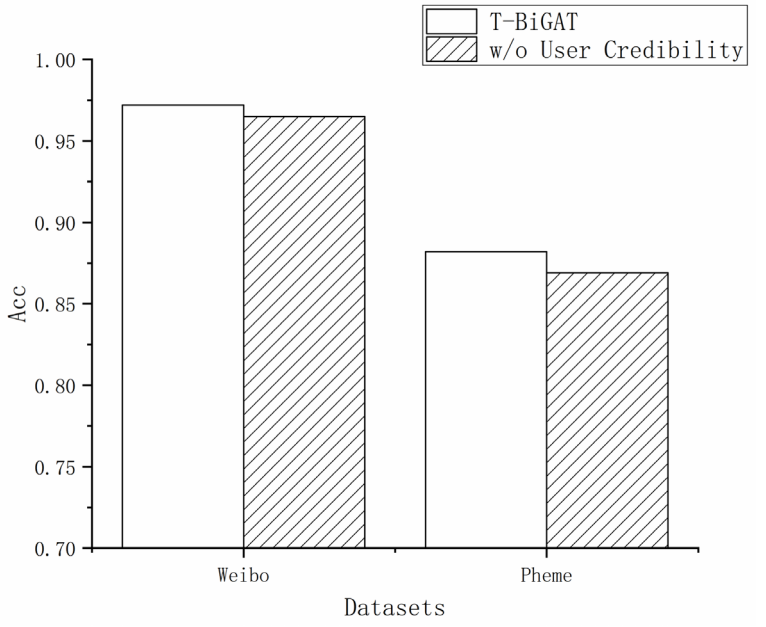


As seen in Figure 4, the rumor detection accuracy of the T-BiGAT model was better than that of the variant model without the introduction of temporal features. In the BiGAT model, the model does not take into account the important factor that tweets posted earlier usually have an impact on tweets posted later and does not capture the temporal features between sibling nodes in a single timestamp subgraph and the relationship between multiple timestamps. In addition, the appearance of subgraphs leads to strong sparseness of graphs, resulting in insufficient node aggregation and poor final detection performance. In the w/o GRU-A model, the model builds multiple subgraphs based on timestamps. Although the problem of tree sparsity still exists, temporal features between tweets are extracted by the GRU model in a single timestamp subtree, which makes the rumor detection performance better than that of the BiGAT model. However, the time correlation of different timestamp subtrees is ignored. Although the w/o GRU-B model and other variant models are based on the construction of subgraphs for feature extraction, the model uses the GRU model to obtain the temporal correlation between different subtrees. Temporal features of tweets are introduced to greatly improve the accuracy of identifying rumors. However, this model does not consider the correlation between sibling nodes, resulting in a slightly lower detection accuracy than T-BiGAT, which introduced temporal features. The T-BiGAT rumor detection model builds subtrees with different timestamps and introduces the GRU model twice to explore the temporal dependencies between tweets. The GRU model can extract temporal features from temporal information, making the T-BiGAT model better than other methods. In summary, it proved the effectiveness of introducing temporal features through the GRU model under the condition of timestamps in this paper.

To explore whether the introduction of user credibility into the T-BiGAT model had a positive impact on the final detection results, the T-BiGAT model was compared with the variant model w/o user credibility, and the accuracy was judged. The experimental results are shown in Figure 5.

- **w/o User Credibility:** This variant model does not introduce user credibility into the node update process of the GAT model.

Figure 5.
Analysis of the Influence of User Credibility on Detection Results



In the public dataset, the T-BiGAT model had higher accuracy in identifying rumors than the baseline method. On the whole, the performance improvement on the Pheme dataset was more obvious after introducing user credibility. In summary, the effectiveness of the T-BiGAT model for introducing user credibility was demonstrated.

Early Rumor Detection

Identifying rumors at an early stage can effectively reduce the harm caused by rumors. This experiment validates the performance of the T-BiGAT model at an early stage by setting a cutoff time of 12 hours on public datasets.

Figures 6 and 7 compare the rumor detection accuracy of each model for different time periods on the two datasets. The rumor detection performance of the T-BiGAT model outperformed the comparative baseline methods within a detection time of 12 hours, and the performance gradually improved as time passed. In the early stage, since the T-BiGAT rumor detection model constructed multiple subgraphs based on timestamps, each subgraph had strong sparsity and insufficient feature extraction, resulting in a low accuracy rate of rumor identification. The number of responding tweets increased over time, and the sparsity of the subtree was largely alleviated. At the 8th hour, the T-BiGAT model was able to effectively extract the temporal features of tweets, and the introduction of user credibility could accurately determine the weight of neighbor nodes, resulting in the model achieving the highest accuracy in the early identification of rumors. In summary, the effectiveness of the T-BiGAT model proposed in this chapter for early rumor detection was verified.

CONCLUSION

This paper proposes a rumor detection model based on a graph attention network with temporal features. To obtain the temporal correlation between tweets, the GRU model is introduced twice to capture the temporal correlation. This model also fuses user credibility with a graph attention network

Figure 6.
Early Rumor Detection Results in the Pheme Dataset

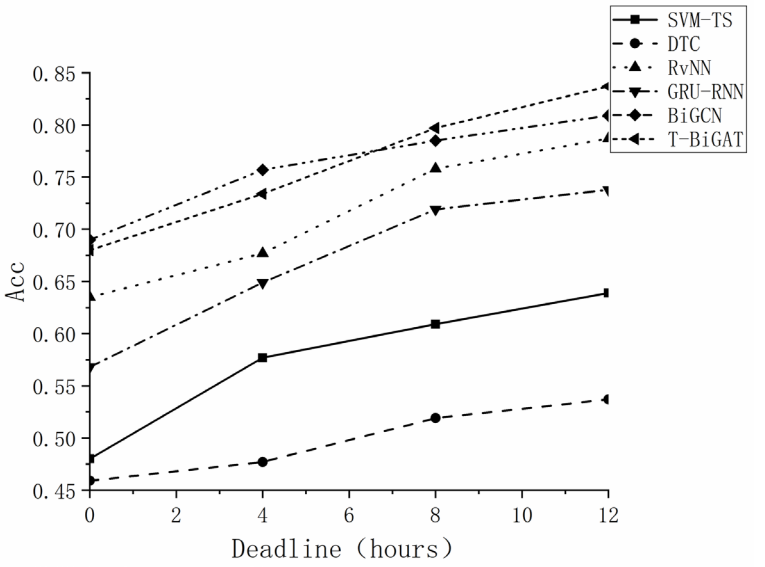
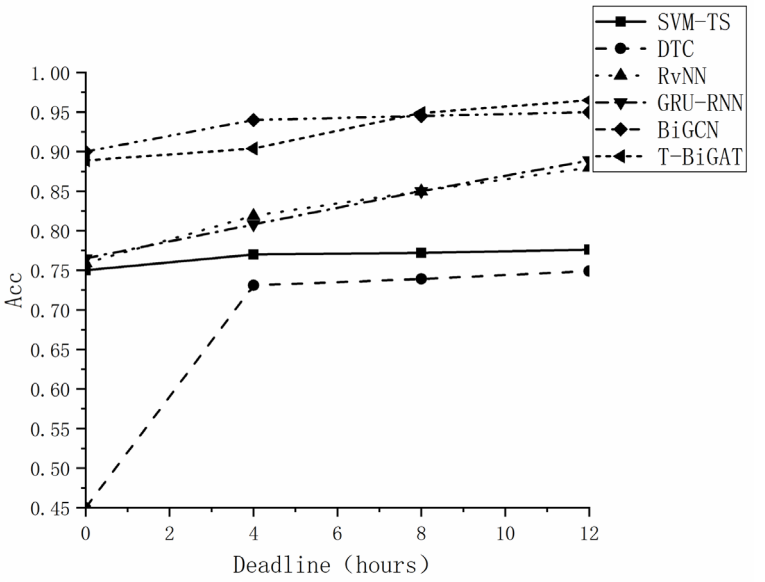


Figure 7.
Early Rumor Detection Results in the Weibo Dataset



to extract features, which effectively selects the credibility of nodes. Experiments show that the model in this paper is better than the baseline methods in each evaluation index. More importantly, this model proves the effectiveness of integrating temporal features and user credibility through ablation experiments. In addition, the method in this paper conducts an early rumor detection performance test by setting a deadline. Experiments showed that the method in this paper has good performance compared to baseline methods in the early stage of rumors.

In future work, first, the labels of response tweets in existing datasets should be annotated; then, information such as images and videos should be aggregated to capture high-order features; finally, due to the shortcomings of graph neural networks, such as difficulty in training, better feature extraction tools should be found for in-depth research.

CONFLICT OF INTEREST

The authors of this publication declare there is no conflict of interest.

FUNDING AGENCY

This research was supported by the National Key R&D Program of China [grant number 2017YFB0802300]; and the Natural Science Foundation of Hebei Province [grant number F2021201052].

REFERENCES

Azri, A., Favre, C., Harbi, N., Darmont, J., & Noûs, C. (2021, September). Calling to CNN-LSTM for rumor detection: A deep multi-channel model for message veracity classification in microblogs. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases* (pp. 497-513). Springer. doi:10.1007/978-3-030-86517-7_31

Bian, T., Xiao, X., Xu, T., Zhao, P., Huang, W., Rong, Y., & Huang, J. (2020, April). Rumor detection on social media with bi-directional graph convolutional networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, *34*(01), 549–556. doi:10.1609/aaai.v34i01.5393

Castillo, C., Mendoza, M., & Poblete, B. (2011). *Information credibility on Twitter*. DBLP. doi:10.1145/1963405.1963500

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning phrase representations using RNN encoder-decoder for statistical machine translation*. doi:10.3115/v1/D14-1179

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). *Bert: Pre-training of deep bidirectional transformers for language understanding*. ArXiv.

Gao, J., Han, S., Song, X., & Ciravegna, F. (2020). *Rp-dnn: A tweet level propagation context based deep neural networks for early rumor detection in social media*. ArXiv.

GuangJun. (2020). Spam detection approach for secure mobile message communication using machine learning algorithms. *Security and Communication Networks*.

Kipf, T. N., & Welling, M. (2016). *Semi-supervised classification with graph convolutional networks*. ArXiv.

Li, B., Qian, Z., Li, P., & Zhu, Q. (2022, June). Multi-modal fusion network for rumor detection with texts and images. In *International Conference on Multimedia Modeling* (pp. 15-27). Springer. doi:10.1007/978-3-030-98358-1_2

Liu, J., Feng, K., Pan, J. Z., Deng, J., & Wang, L. (2020). MSRD: Multimodal web rumor detection method. *Journal of Computer Research and Development*, *58*(7), 1395–1411.

Lotfi, S., Mirzarezaee, M., Hosseinzadeh, M., & Seydi, V. (2021). Detection of rumor conversations in Twitter using graph convolutional networks. *Applied Intelligence*, *51*(7), 4774–4787. doi:10.1007/s10489-020-02036-0

Ma, J., Gao, W., Mitra, P., Kwon, S., & Cha, M. (2016). Detecting rumors from microblogs with recurrent neural networks. *International Joint Conference on Artificial. Intelligence. Proceedings of the 25th International Joint Conference on Artificial Intelligence (IJCAI 2016)*, 3818-3824.

Ma, J., Gao, W., Wei, Z., Lu, Y., & Wong, K. F. (2015, October). Detect rumors using time series of social context information on microblogging websites. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management* (pp. 1751-1754). ACM. doi:10.1145/2806416.2806607

Ma, J., Gao, W., & Wong, K. F. (2018). *Rumor detection on Twitter with tree-structured recursive neural networks*. Association for Computational Linguistics. doi:10.18653/v1/P18-1184

Ma, Y., Xu, S., & Dong, F. (2022, January). A multilevel graph convolution neural network model for rumor detection. In *2022 IEEE 2nd International Conference on Power, Electronics and Computer Applications (ICPECA)* (pp. 1225-1229). IEEE. doi:10.1109/ICPECA53709.2022.9719043

Min, H. E., Jie, X. U., Pan, D. U., Cheng, X. Q., & Wang, L. H. (2016). Bursty topic detection method for microblog based on time series analysis. *Journal of Communication*, *37*(003), 48–54.

Shelke, S., & Attar, V. (2022). Rumor detection in social network based on user, content and lexical features. *Multimedia Tools and Applications*, *81*(12), 17347–17368. doi:10.1007/s11042-022-12761-y PMID:35282405

Shi, L., Du, J., & Liang, M. (2018). Social network bursty topic discovery based on RNN and topic model. *Journal of Communication*, *39*(4), 189–198.

Xu, G., Zhou, D., & Liu, J. (2021). Social network spam detection based on ALBERT and combination of Bi-LSTM with self-attention. *Security and Communication Networks*, *2021*, 2021. doi:10.1155/2021/5567991

Xue, H., Wang, L., Yang, Y., & Lian, B. (2021). Rumor detection model based on user propagation network and message content. *Jisuanji Yingyong*, *41*(12), 3540–3545.

Zubiaga, A., Liakata, M., Procter, R., Hoi, G., & Tolmie, P. (2016). Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLoS One*, *11*(3), e0150989. doi:10.1371/journal.pone.0150989 PMID:26943909