Rumor Detection on Social Media with Graph Adversarial Contrastive Learning

Tiening Sun
Soochow University
Suzhou, China
tnsun@stu.suda.edu.cn

Zhong Qian
Soochow University
Suzhou, China
qianzhong@suda.edu.cn

Sujun Dong
Soochow University
Suzhou, China
sjdong@stu.suda.edu.cn

Peifeng Li
Soochow University
Suzhou, China
pfli@suda.edu.cn

Qiaoming Zhu*
Soochow University
Suzhou, China
qmzhu@suda.edu.cn

ABSTRACT

Rumors spread through the Internet, especially on Twitter, have harmed social stability and residents' daily lives. Recently, in addition to utilizing the text features of posts for rumor detection, the structural information of rumor propagation trees has also been valued. Most rumors with salient features can be quickly locked by graph models dominated by cross entropy loss. However, these conventional models may lead to poor generalization, and lack robustness in the face of noise and adversarial rumors, or even the conversational structures that is deliberately perturbed (e.g., adding or deleting some comments). In this paper, we propose a novel Graph Adversarial Contrastive Learning (GACL) method to fight these complex cases, where the contrastive learning is introduced as part of the loss function for explicitly perceiving differences between conversational threads of the same class and different classes. At the same time, an Adversarial Feature Transformation (AFT) module is designed to produce conflicting samples for pressurizing model to mine event-invariant features. These adversarial samples are also used as hard negative samples in contrastive learning to make the model more robust and effective. Experimental results on three public benchmark datasets prove that our GACL method achieves better results than other state-of-the-art models.

CCS CONCEPTS

• **Computing methodologies** → *Natural language processing*; *Machine learning*; • **Information systems** → *World Wide Web.*

KEYWORDS

rumor detection, contrastive learning, adversarial learning, graph representation, social networks

ACM Reference Format:
Tiening Sun, Zhong Qian, Sujun Dong, Peifeng Li, and Qiaoming Zhu. 2022. Rumor Detection on Social Media with Graph Adversarial Contrastive Learning. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3485447.3511999

1 INTRODUCTION

The convenient feature of social media has accelerated the spread of false or unverified information on a large scale in the social network like a virus, which seriously disturbs the health of the network environment and degrades the user experience. Misinformation (especially malicious rumors) can mislead the public, affect personal life and normal social stability, and even directly have a profound impact on financial markets and national politics. Hence, it is urgent to construct an effective rumor detection method.

Deep learning plays an important role in rumor detection, which can automatically and efficiently learn the feature vectors containing deep semantic information from the text, pictures and structure of rumors [13]. For example, the Recurrent Neural Network (RNN) represented by Long Short Term Memory (LSTM) and Gate Recurrent Unit (GRU), and its various variants, can effectively capture the time series relationship between each post in the rumor propagation chain [16, 30]. Convolutional Neural Network (CNN) based methods have the ability to learn the local spatial feature representation [15, 35]. However, these methods only focus on the text information of the rumors and ignore the structural information of rumor propagation. Thus, in order to get closer to reality, some studies have tried to incorporate the propagation structure information into the rumor detection model by invoking Graph Neural Network (GNN) based methods [1, 15, 27, 36].

Despite GNN's success in rumor detection, the aforementioned methods that use cross entropy loss function often lead to a poor generalization capability [14] and a lack of robustness against noise [37], and adversarial samples shown in Figure 1, especially malicious rumors [32]. Sometimes, just setting a simple perturbation can cause labels to be misclassified with a high degree of confidence, which is undoubtedly a huge potential harm for the rumor classification system. Hence, existing data-driven models need to become more robust in the face of misinformation usually generated and spread by normal users unconsciously, and the confusing dialogue

[*] Corresponding Author.

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3511999

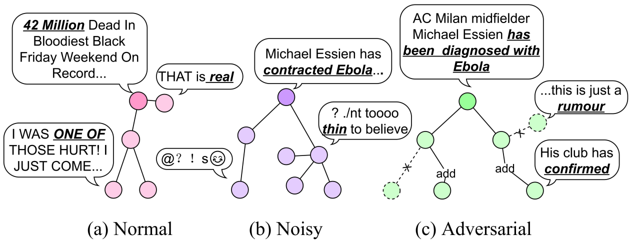
42 Million Dead In Bloodiest Black Friday Weekend On Record...
THAT is real
I WAS ONE OF THOSE HURT! I JUST COME...
@ ? ! s☺
Michael Essien has contracted Ebola...
? ./nt toooo thin to believe
AC Milan midfielder Michael Essien has been diagnosed with Ebola
...this is just a rumour
His club has confirmed
add
add
(a) Normal
(b) Noisy
(c) Adversarial

Figure 1: (a) A normal conversational thread graph (or called rumor propagation tree), where each graph node represents a post, and the edge represents the comment relationship between two posts; (b) A noisy graph, which contains some noise such as misplaced comments, spelling errors, grammatical confusion, garbled characters, etc; (c) An adversarial example in which a malicious user deliberately deletes some unfavorable comments and adds comments that are beneficial to his post.

structures maliciously designed by rumor producers, which leads to our innovations below.

In order to achieve more robust and effective detection, we propose a Graph Adversarial Contrastive Learning (GACL) method for rumor detection in this paper, which is inspired by the learning strategy that humans not only capture similarities between examples in one class, but also compare them to examples in other classes when classifying a target. Specifically, we first adopt the graph data enhancement strategies such as edge perturbation and dropout mask to simulate the case of Figure 1(b), which provides input data with rich noise for the model. Then, we introduce supervised graph contrastive learning [10] shown in Figure 2 to train our GNN encoder to explicitly perceive the differences in the augmented data, and learn robust representation. Unlike the self-supervised contrastive learning strategies, our method can utilize the label information more efficiently. In this way, we can prevent some cases containing noise such as misplaced comments and garbled characters from being misclassified by the detection model.

Sometimes this alone is not enough. Because in the real world, in addition to misinformation unintentionally created and spread by normal users, there are also malicious rumors carefully designed and deliberately promoted by rumor producers as shown in Figure 1(c), which may disable the model. Some researchers have also paid attention to this issue. Ma et al. [21] analyzes a rumor case about "Saudi Arabia beheads first female robot citizen" to illustrate how rumor bots use high-frequency and indicative words to cover up the facts. Yang et al. [32] also mention that rumors producers often manipulate the relationship network composed of users, sources and comments to escape detection. Whether text tampering or network manipulation, the purpose of the rumor producers is to make the rumors close to the non-rumor samples in the high-dimensional space, thereby confusing the model. Hence, in order to solve this problem, we develop an Adversarial Feature Transformation (AFT) module, which aims to utilize adversarial training to generate challenging features. These adversarial features will be used as hard negative samples in contrastive learning to help the model strengthen feature learning from these difficult samples, and achieve robust and effective detection. In addition, we intuitively believe that these adversarial features can be decoded into a wide range of various types of perturbations.

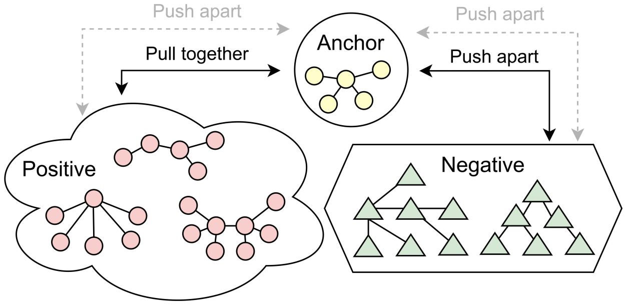

Figure 2: An illustration case of supervised graph contrastive learning. Taking a given conversational thread graph as an anchor, the proposed framework attempts to pull the anchor and positive instances belonging to the same class together, while pushing away negative samples that do not belong to the same class, as shown by the solid line with arrows. The dotted line with arrows points out the common self-supervised contrastive learning strategy without considering label information, that is, pushing away all other instances except the augmented self.

Our contributions are summarized as follows:

- To the best of our knowledge, this is the first study to introduce contrastive learning into the rumor detection task, which aims to improve the quality of representation by perceiving the differences between samples of the same label and different labels.
- We propose the GACL model that not only considers the propagation structure information of rumors but also simulates noise and adversarial cases, and captures the event-invariant features by utilizing contrastive learning.
- Under the GACL framework, we develop the AFT module to generate adversarial features that are used as hard negative samples in contrastive learning to learn more robust representations.
- We experimentally demonstrate that our model outperforms state-of-the-art baselines on real-world datasets.

2 RELATED WORK

In this section, we review the related research work of rumor detection on social media, and briefly introduce the current researches of contrastive learning.

2.1 Rumor Detection Methods

The research on rumors detection mostly revolves around extracting the content information of the rumors, summarizing various statistical characteristics, and analyzing the propagation path of the rumors. Recently, deep learning models have achieved state-of-the-art performance in rumor detection tasks, which can automatically mine potential semantic information while overcoming the shortcomings of hand-crafted features. Ma et al. [16] introduced the RNN to capture potential temporal semantic information and successfully defeated rumor detection models that use hand-crafted features. Yu et al. [35] utilized CNN to extract sequence features and shaped high-level interactions among key features. Ma et al. [21] proposed a method based on Generative Adversarial Networks (GAN) to capture low-frequency but stronger non-trivial patterns and improve the robustness of algorithm. In addition, many previous studies have shown that user stance, user credibility and multi-modal (textual + visual) information of rumors are very important in rumor detection [8, 12, 13, 19, 23, 25, 28].

The common limitation of aforementioned methods is that they do not fully consider the network and propagation structure of rumors. Ma et al. [20] constructed a bottom-up and a top-down tree-structured neural network for rumor detection on Twitter. Bian et al. [1] improved this approach by adopting a more sophisticated GCN. Yuan et al. [36] built a heterogeneous information network involving users, sources and comments for rumor detection. And on this basis, four kinds of camouflage behaviors were designed to improve the robustness of the model [32]. Our framework, an upgraded version of Bian et al. [1]'s approach, encourages the model to capture the similarities between rumors and compare them with non-rumors, which is beneficial to learning invariant features of each class and achieving more robust and effective detection.

2.2 Contrastive Learning

Contrastive learning, whose core idea is to learn from positive samples and benefit from correcting negative ones, has been successfully applied to many tasks. For example, Dai and Lin [5] employed the contrastive learning for image caption. Chen et al. [3] used contrastive learning to improve the quality of the visual representations. Wu et al. [29] proposed to evaluate the summary qualities by unsupervised contrastive learning. Cai et al. [2] introduced contrastive learning into dialogue generation to improve the diversity of responses. In addition, the contrastive learning has successfully promoted the development of representation learning of graph-structured data. Different graph data enhancement schemes are successively proposed, with the purpose of further obtaining a generalizable, transferable and robust graph representation [24, 34, 38]. Our work is inspired by self-supervised contrastive learning, but the difference is that we develop a supervised graph contrastive learning classifier specifically for the rumour detection task, which also makes full use of the hard negative samples generated by adversarial training.

3 PROBLEM DEFINITION

Rumor detection is defined as a classification task whose purpose is to learn a classifier from a set of labeled training events, and then use it to predict the label of the test event in this paper. Specifically, we represent the event set as $C = \{c_1, c_2, ..., c_n\}$, where c_i is i-th event and n is the number of events. Each event $c = (y, G)$ consists of the ground-truth label $y \in \{R, N\}$ of the event (i.e. Rumor or Non-rumor) and the graph $G = (V, E)$ referring to propagation structure, where V is the set of graph nodes and E is the set of edges. In some cases, rumor detection is defined as a four-class classification task, correspondingly, $y \in \{N, F, T, U\}$ (i.e., Non-rumor, False Rumor, True Rumor, and Unverified Rumor). Moreover, in the model training stage, $\hat{G}$ is generated through data enhancement, whose goal is to learn a classifier $f(\cdot)$ together with the original G. But in the testing stage, only the original G is used to predict the label of a given event c_i.

4 METHOD

In this section, we propose a supervised GACL method for rumor classification tasks, which attempts to achieve robust and effective detection in the face of noise and adversarial data (especially samples that have been maliciously perturbed by rumor producers) on social media. As shown in Figure 3, we will elaborate the process of using GACL to classify rumors, including graph data augmentation, graph representation, AFT component, rumor classification and adversarial contrastive Learning.

4.1 Graph Data Augmentation

For GACL, graph data augmentation is a prerequisite, which aim at creating realistically new data by performing some conversion without affecting semantic labels. In this work, the *edge perturbation* strategy is employed to perform the data augmentation. Specifically, given a graph $G = (V, E)$ with the adjacency matrix A and the feature matrix X, edge perturbation will randomly dropping, adding or misplacing some edges with probability r in each training epoch (as shown by the instances G in Figure 3) to perturb the connectivities of G. Formally, suppose the newly generated graph data is named $\hat{G}$, and $A_{perturbation}$ is the matrix constructed using edges randomly sampled from the original edge set, then the adjacency matrix $A^{'}$ of $\hat{G}$ can be computed as $A^{'} = A - A_{perturbation}$. $\hat{G}$ has certain robustness to the situation of edge connectivity pattern variances, such as facing the camouflage structures designed by rumor producers.

In addition, for the rumor detection task, the text information of graph nodes composed of the posts in Figure 3 is also one of the key clues to correctly classify rumors [1], which should also be augmented to provide some noises. Gao et al. [7] recently discovered that model can achieve state-of-the-art performance by only applying the *dropout mask*. For simplicity, we only need to randomly mask a small number of neurons or words in each post for rich noise, as shown in Figure 3.

4.2 Graph Representation

Given an input data G, after the graph data augmentation operation, the correlated view $\hat{G}_k$ is obtained, which contains some subset of the information in the original sample. Correspondingly, the adjacency matrix A from G will be converted into $A^{'}_{k}$ as the edges are dropped, added or misplaced with probability r in each training epoch. For the text information of the graph nodes, we use dropout mask operation to generate the noisy text samples with a small amount of missing information, and employ the Bidirectional Encoder Representations from Transformers (BERT) [22] which has been successfully applied in fields such as classification [26], translation, etc., to separately encode the source and comments to form new feature matrix X_k. In order to emphasize

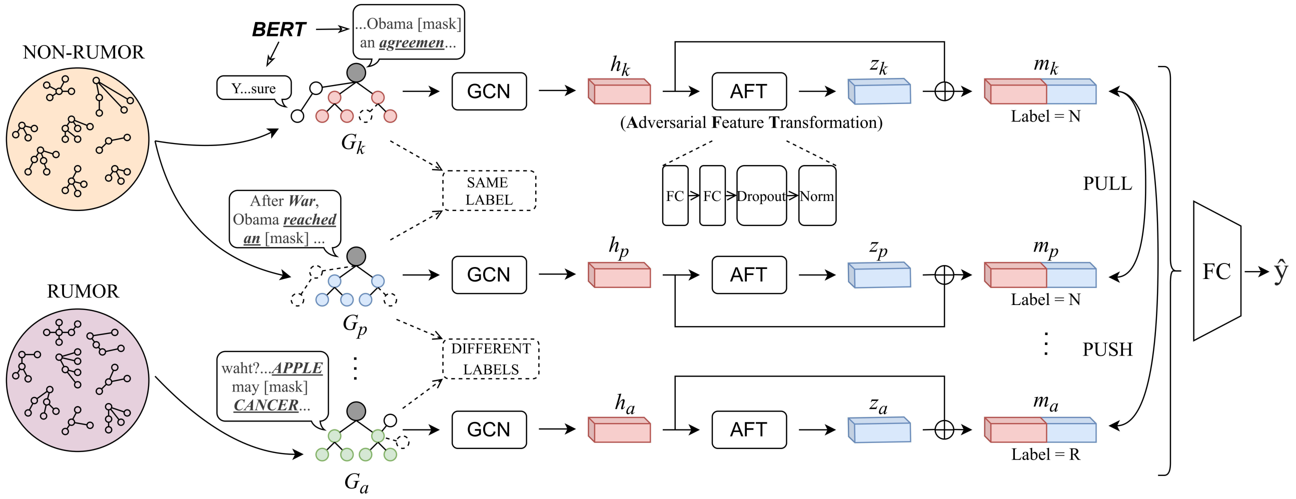

Figure 3: Overview of our GACL rumor detection model. Given an input batch of data, we split it into two clusters referring to rumors and non-rumors. Then, the various types of data augmentation strategies are applied to generate the perturbed rumor trees such as G_k, G_p and G_a (that are just generic elements and used as examples). Next, the representation of the rumor trees is calculated using BERT and GCN to obtain a 64-dimensional feature vector h. Finally, h and the adversarial feature z generated by the AFT module are concatenated together for subsequent contrastive training and classification.

the importance of the source post (that is, the content information of the root node), we join the source post and comment in a [*CLS*] *Source* [*SEP*] *Comment* [*SEP*] manner, and the final hidden state representation of [*CLS*] token is used as its corresponding node representation. The popular Graph Convolutional Network (GCN) [11] has shown superior capability in aggregating graph information, which is then employed in our work to obtain high-quality graph representation.

Specifically, first each node in the augmented graph $\hat{G}_k$ is added with the self-connection, in consequence the new adjacency matrix $\tilde{A}_k$ is expressed as

$$\tilde{A}_k = A'_k + I, \tag{1}$$

Then, we feed data into GCN, whose forward propagation process can be formulated as

$$H^{(l+1)} = \sigma(\hat{A}H^{(l)}W^{(l)}), \tag{2}$$

where σ is an activation function such as the ReLU function. $\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}D^{-\frac{1}{2}}$ is the normalized adjacency matrix, where $\tilde{D}$ is $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ that represents the degree. $W^{(l)}$ is the weight matrix. $H^{(l)}$ represents the hidden feature of l-th layer. $H^{(0)} = X$, where X is the initial feature matrix. In our work, two-layer GCN is employed. The forward propagation process is formulated as

$$H_k^{(2)} = \sigma(\hat{A}_k\sigma(\hat{A}_kXW_k^{(0)})W_k^{(1)}), \tag{3}$$

where the ReLU function is adopted as activation function σ, and $\hat{A}_k = \tilde{D}^{-\frac{1}{2}}\tilde{A}_kD^{-\frac{1}{2}}$. W_k is the trainable parameter matrix.

Finally, we use mean-pooling operators (MEAN) to aggregate the information of $H_k^{(2)}$ representing the set of node representations. It is formulated as

$$h_k = MEAN(H_k^{(2)}), \tag{4}$$

4.3 AFT Component

Even if the AFT module does not exist, the graph representation h generated by GCN can be directly fed into the final softmax layer for rumor classification. However, since the model has only been exposed to the input samples generated by data augmentation that contain random noise during the training phase, it lacks robustness to adversarial samples (especially some data that is carefully perturbed by humans), as shown in Figure 1(c). In order to evade model detection, rumor producers may use graph camouflage strategies to make the conversational threads closer to the non-rumor instances, thereby confusing the graph detection model [32]. They may also utilize rumor bots to post lots of comments that contain many high-frequency and indicative words to cover up the facts [21]. The ultimate goal of these cases is to make the rumor feature vectors closer to the non-rumor feature vectors in the latent space. The proposed AFT module based on adversarial learning attempts to simulate these behaviors in a high-dimensional space, and generate adversarial vectors for pressurizing model to mine event-invariant features in the training phase.

As shown in Figure 3, the AFT is composed of a stack of $L = 2$ fully connected layers, Dropout and Normalization (DN). After passing through the AFT module, h_k are converted into z_k, which is formulated as

$$z_k = DN(max(0, h_kW_1^{AFT} + b_1)W_2^{AFT} + b_2), \tag{5}$$

where W^{AFT} and b are the weight matrix and bias respectively. The parameters of the AFT module are trained using adversarial learning, and the obtained z_k vector will be used as the hard negative sample in the contrastive learning, which is introduced in detail in the Adversarial Contrastive Learning section.

4.4 Rumor Classification

Now, for each post in a batch, we have obtained the corresponding graph representations h_k encoded by GCN, and adversarial representations z_k generated by AFT. Then, we concatenate them to merge the information as

$$m_k = concat(h_k, z_k), \tag{6}$$

Next, m_k is fed into full-connection layers and a softmax layer, and the output is calculated as

$$\hat{y}_k = softmax(W_k^F m_k + b_k^F), \tag{7}$$

where $\hat{y} \in \mathbb{R}^{1\times C}$ is the predicted probability distribution. W^F and b^F are the trainable weight matrix and bias respectively.

4.5 Adversarial Contrastive Learning

We construct a novel loss function as the optimization objective for the supervised rumor classification, which aims to maximize the consistency between the positive examples pairs while pushing away the negative examples, given the labels. Specifically, taking m_k in Figure 3 as the anchor, the m_p with the same label as the anchor m_k is regarded as a positive sample, and the m_a with a different label from the anchor is regarded as a negative sample. Assuming that the label of m_k is the non-rumor at the moment, we try to increase the cosine similarity between m_k and each non-rumor vector in the high-dimensional space, while reducing the similarity with rumor vectors, so as to help model learn the event-invariant representation. Unlike the application of contrastive learning to self-supervised tasks [7, 24], we effectively leverage label information and incorporate cross-entropy into the optimization objective, so that the model can quickly improve the ability of rumor classification during the training phase. Hence, the final loss function will contain two parts: cross entropy loss and contrastive learning loss [10, 33], which can be calculated as follows

$$\mathcal{L} = \mathcal{L}_{ce} + \alpha\mathcal{L}_{sup}, \tag{8}$$

where

$$\mathcal{L}_{ce} = -\frac{1}{N}\sum_{k=1}^{N}\sum_{c=1}^{M} y_{k,c} log(\hat{y}_{k,c}), \tag{9}$$

$$\mathcal{L}_{sup} = -\sum_{k\in K} log\left\{\frac{1}{|P(k)|}\sum_{p\in P(k)}\frac{exp(sim(m_k, m_p)\tau)}{\sum\limits_{a\in A(k)} exp(sim(m_k, m_a)\tau)}\right\}, \tag{10}$$

and α is the adjustable hyperparameter. In $\mathcal{L}_{ce}$, $y_{k,c}$ denotes ground-truth label and $\hat{y}_{k,c}$ denotes the predicted probability distribution of index $k \in K \equiv \{1...N\}$ belonging to class $c \in \{1...M\}$. In $\mathcal{L}_{sup}$, the index k is called as *anchor*, index p corresponds to the positive sample with the same label as the anchor k, and the index a corresponds to the negative sample with the label different from the anchor k. $A(k) \equiv \{a \in K : y_a \neq y_k\}$ is the set of indices of negative samples in the minibatch, and $P(k) \equiv \{p \in K : y_p = y_k\}$ is the set of indices of positive samples. $sim(\cdot)$ denotes the cosine similarity function such as $sim(m_k, m_p) = m_k^T m_p / \|m_k\|\|m_p\|$, and $\tau \in \mathcal{R}^+$ is a scalar temperature parameter.

Algorithm 1 Adversarial contrastive training procedure.

```
Input: A set of input graphs $G$, learning rate $\epsilon$
Parameter: $\theta_a$, $\theta_s$
Initialize $\theta_a$ and $\theta_s$ with random weight values;
for epoch from 1 to maxEpoch do
    for each mini-batch of $G$ do
        Generate copies $\hat{G}$ using data augmentation;
        Calculate graph representation using Equation (3);
        Calculate adversarial representation using Equation (5);
        Obtain fusion feature vector using Equation (6);
        Compute loss $\mathcal{L}$ using Equation (8);
        /* Minimize $\mathcal{L}$ w.r.t. $\theta_s$ */
        Compute gradient $\nabla(\theta_s)$;
        Update $\theta_s$: $\theta_s \leftarrow \theta_s - \epsilon\nabla(\theta_s)$;
        /* Maximize $\mathcal{L}$ w.r.t. $\theta_a$ */
        Compute gradient $\nabla(\theta_a)$;
        Update $\theta_a$: $\theta_a \leftarrow \theta_a + \epsilon\nabla(\theta_a)$
    end for
end for
```

Moreover, some studies have shown that the BERT-driven sentence representations that will be applied to our work are collapsed [4], where the semantic information of the sentence is dominated by high-frequency words [31]. In rumor detection, high-frequency and indicative words are often utilized by rumor producers to confuse the detection model [21]. Hence, invoking contrastive learning can smooth the sentence semantics and increase the weight of low-frequency but strong words theoretically. Finally, we minimize the loss function $\mathcal{L}$ to update the model parameters, except for AFT.

Adversarial Training: The AFT module is trained separately based on adversarial learning. Assume that the parameter of the AFT module in GACL is θ_a, and the parameters of the remaining modules are θ_s. In each epoch, we first minimize $\mathcal{L}$ to update parameter θ_s, and then maximize $\mathcal{L}$ to update parameter θ_a. We utilize adversarial learning to minimize the consistency between the adversarial sample and samples with the same label, and maximize the similarity between it and samples with different labels, so as to achieve the purpose of confusing the model. The detailed training process is shown in Algorithm 1.

5 EXPERIMENTS

In this section, we first evaluate the proposed GACL method by comparing it with some benchmark models, and give some discussion and analysis. Second, we perform ablation analysis to verify the effectiveness of each module of GACL in turn. Finally, we evaluate the ability of GACL in the early rumor detection task.

5.1 Datasets

We evaluate the proposed GACL model on three public real-world datasets: Twitter15 [18], Twitter16 [18] and PHEME [39], all of which are collected from Twitter that is the most influential social

Table 1: Statistics of the datasets

Statistic	Twitter15	Twitter16	PHEME
# source tweets	1490	818	6425
# non-rumors	374	205	4023
# false rumors	370	205	2402
# unverified rumors	374	203	-
# true rumors	372	205	-
# users	276,663	173,487	48,843
# posts	331,612	204,820	197,852

media site in the US. There are two versions of PHEME, which are collected based on five and nine breaking news respectively. In our work, the version containing nine events is selected. Both Twitter15 and Twitter16 contain four tags: Non-rumor (N), False Rumor (F), True Rumor (T), and Unverified Rumor (U), which are used for quaternary classification. PHEME contains only two types of tags: Rumor (R) and Non-Rumor (N), which is used for the binary classification of rumors and non-rumors. Furthermore, graph topologies of posts are constructed based on users, sources and comments in the three datasets, where the text content contained in each graph node is represented by BERT and the graph structure is encoded by GCN. Detailed statistics are shown in Table 1.

5.2 Experimental Settings

We make comparisons with the following state-of-the-art baselines:

SVM-TS [17] is a linear SVM classifier that can use handcrafted features to capture the variation of social context features.

CNN [35] is a CNN-based model that can learn the local spatial features between rumor posts.

BERT [6] is a popular pre-trained model that is used for rumor detection.

RvNN [20] is a tree-structured rumor classifier that can extract high-level representations by analyzing the bottom-up and top-down propagation tree.

GCAN [15] is a GCN-based model that can describe the rumor propagation mode and use the dual co-attention mechanism to capture the relationship between source text, user characteristics and propagation path.

UDGCN [1] directly uses GCN for rumor detection, in which the root feature enhancement strategy is used to improve the performance of the model.

BiGCN [1] is a GCN-based model that uses the two key features of rumor propagation and dispersion to capture the global structure of the rumor tree.

GACL(our) is a GCN-based model using adversarial and contrastive learning, which can not only encode the global propagation structure, but also resist noise and adversarial samples, and capture invariant features.

The proposed GACL [1] model is implemented by PyTorch [9]. As with BiGCN, we randomly split the dataset into five parts and construct 5-fold cross-validation. At the same time, the Accuracy (Acc.), Precision (Prec.), Recall (Rec.) and F_1-measure (F_1) are adopted as evaluation metrics in all three datasets. In addition, the learning rate is initialized to 5e-4 and gradually decreases during training according to the decay rate of 1e-4. The temperature parameter is set to 0.3, 0.3 and 0.6, respectively, for the Twitter15, Twitter16 and PHEME datasets.

Table 2: Rumor detection results on Twitter15 dataset

Twitter15					
Method	Acc.	N	F	T	U
		F_1	F_1	F_1	F_1
SVM-TS	0.642	0.811	0.434	0.639	0.600
CNN	0.718	0.807	0.601	0.635	0.730
RvNN	0.723	0.682	0.758	0.821	0.654
BERT	0.735	0.731	0.722	0.730	0.705
GCAN	0.842	0.844	0.846	0.889	0.800
UDGCN	0.834	0.827	**0.866**	0.885	0.756
BiGCN	0.886	0.891	0.860	**0.930**	0.864
GACL	**0.901**	**0.958**	0.851	0.903	**0.876**

Table 3: Rumor detection results on Twitter16 dataset

Twitter16					
Method	Acc.	N	F	T	U
		F_1	F_1	F_1	F_1
SVM-TS	0.691	0.763	0.483	0.722	0.690
CNN	0.700	0.688	0.666	0.810	0.615
RvNN	0.737	0.662	0.743	0.835	0.708
BERT	0.804	0.777	0.525	0.824	0.787
GCAN	0.871	0.857	0.688	0.929	0.901
UDGCN	0.867	0.789	0.846	0.903	0.878
BiGCN	0.880	0.847	0.869	0.937	0.865
GACL	**0.920**	**0.934**	**0.869**	**0.959**	**0.907**

Table 4: Rumor detection results on PHEME dataset

PHEME					
Method	Class	Acc.	Prec.	Rec.	F_1
SVM-TS	R	0.685	0.553	0.539	0.539
	N		0.758	0.762	0.757
CNN	R	0.747	0.683	0.512	0.584
	N		0.768	0.872	0.816
RvNN	R	0.763	0.689	0.587	0.631
	N		0.796	0.858	0.825
BERT	R	0.807	0.736	0.695	0.713
	N		0.842	0.866	0.853
GCAN	R	0.834	0.769	**0.758**	0.761
	N		0.871	0.874	0.872
UDGCN	R	0.805	0.752	0.673	0.708
	N		0.831	0.875	0.852
BiGCN	R	0.824	0.753	0.734	0.741
	N		0.861	0.872	0.865
GACL	R	**0.850**	**0.801**	0.750	**0.772**
	N		**0.871**	**0.901**	**0.885**

[1] The code will be available at https://github.com/agangbe/GACL

5.3 Results and Discussion

Tables 2, 3 and 4 show the performance of all comparison methods on three public real-world datasets, where the bold part represents the best performance (our model GACL is significantly superior to the other baselines with a p-value < 0.026). The results show that the proposed GACL model outperforms all baselines, which confirms the advantages of introducing adversarial contrastive learning and graph model into the supervised rumor detection task.

Unsurprisingly, the SVM-TS based on low-level hand-crafted features get the worst results. CNN and RvNN based on deep learning obtain moderate test results. CNN only considers local spatial features, while RvNN can encode global structure information by analyzing top-down and bottom-up propagation relationships in rumor trees, so RvNN has better performance. The BERT model with a self-attention mechanism can generate better text representations of posts, which helps to improve prediction accuracy.

GCAN, UDGCN and BiGCN are all GCN-based models, which are state-of-the-art benchmarks used to verify the superiority of GACL proposed in this paper. These three models show excellent performance. GCAN uses dual co-attention mechanism to mine the relationship between rumor propagation structure, user characteristics and context information. UDGCN and BiGCN mainly rely on powerful GCN encoder to capture the global structure features of rumor trees. Compared with the UDGCN model, the average accuracy of BiGCN on the three datasets is improved by 3% by fusing the bottom-up and top-down structure information of rumors.

The GACL proposed in this paper beats all benchmarks, whether it is tested on the Twitter15 and Twitter16 containing four classes, or on the PHEME containing two classes [2] with an unbalanced number of instances. Compared with GCAN, UDGCN and BiGCN, the average accuracy of GACL on the three data sets is improved by 4%, 6% and 3% respectively. The superiority of GACL stems from four reasons:

1) GCN is adopted to encode the natural topology structure between rumor posts, while BERT, an advanced pre-training model that can dynamically adjust word embedding according to the context to solve the phenomenon of polysemy, is used to encode the text information of each graph node.
2) The flexible data enhancement strategy is adopted. Specifically, by masking part of the text words and randomly deleting or adding some edges of the rumor tree, noise samples are generated and fed to the model as input in the training phase, which has certain robustness to the situations of misplaced comments, spelling errors and so on.
3) The introduction of contrastive loss on the basis of the original cross-entropy loss can help the model better learn the commonalities between augmented samples of the same class and the differences between samples of different classes, thereby generating high-quality feature representations.
4) In the real world, in addition to noise samples, there are some artificially perturbed samples (or called camouflage [32]), as shown in Figure 1(c). Obviously, the common data enhancement and contrastive learning cannot handle such adversarial samples. There are so many camouflage patterns that we can't see through them one by one, but their ultimate purpose is to approximate non-rumor samples in a high-dimensional space to achieve the effect of escaping the detection model. The adversarial features generated by the AFT module based on adversarial training can simulate the similar situations to achieve robust detection. At the same time, the adversarial features are used as hard negative samples in contrastive learning, increasing the difficulty in learning. Adversarial learning and contrastive learning are mutually reinforcing.

Table 5: Results of ablation study on the Twitter15, Twitter16 and PHEME

Model	Twitter15	Twitter16	PHEME
	Acc.	Acc.	Acc.
GACL	0.901	0.920	0.850
GACL-TEXT	0.868	0.899	0.815
GACL-NT	0.882	0.893	0.841
GACL-NCL	0.876	0.907	0.842

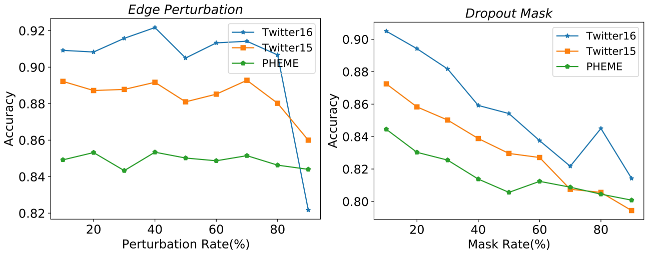

Figure 4: The impact of varying the perturbation rate of edges and applying dropout mask with different rates.

In addition, the prediction results in PHEME are significantly inferior to Twitter 15 and Twitter 16. It is because that the content information of the posts in PHEME is only based on nine events, and there is a lot of overlap in language description. At the same time, the average number of comments per source in PHEME is only 30, while the average number of comments per source is 233 and 250 in Twitter 15 and Twitter 16. Hence, our GACL model cannot capture commonalities and differences from more events due to too little information.

5.4 Ablation Study

In order to verify the effectiveness of the different modules of GACL, we compare it with the following variants:

GACL-TEXT only uses the text information including the source and comments, and does not consider the structure information of the rumor trees.

GACL-NT removes the AFT module, which makes the model lose the ability of generating adversarial feature vectors.

GACL-NCL removes the contrastive learning loss, making the model unable to capture differences between instances of different classes and reducing the quality of representations.

[2] Following previous work on PHEME, we also conduct two-class (i.e., Rumor(R) and NonRumor(N) where Rumor contains Non-rumor, False Rumor and True Rumor) classification.

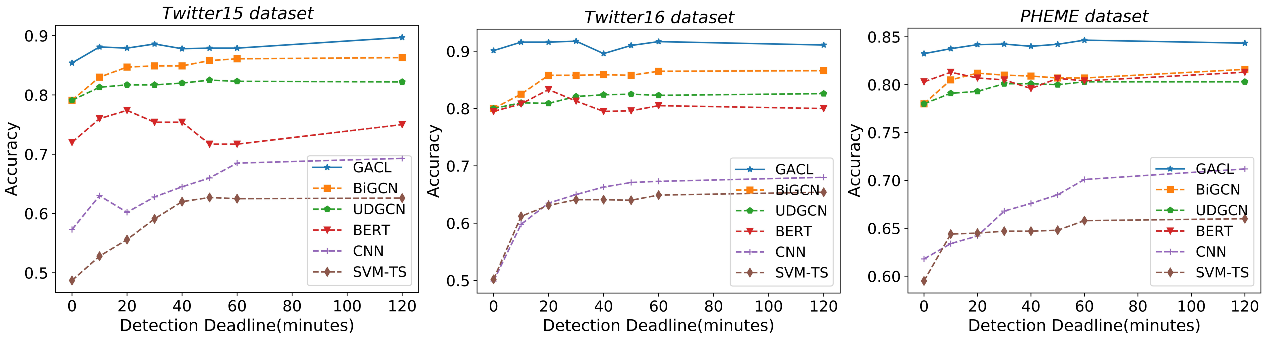

Figure 5: Results of rumor early detection on three datasets.

The experimental results are shown in Table 5. It can be observed: 1) Compared with GACL, the accuracy of GACL-TEXT on the Twitter15, Twitter16 and PHEME datasets is reduced by 3.3%, 2.1% and 3.5%, respectively. Obviously, when contrastive learning is directly applied to a long text in which the source and comments are simply concatenated, the model only can achieve a moderate prediction accuracy due to the chaotic textual pattern and the lack of conversational thread structure. 2) The lack of AFT module will reduce the overall performance of GACL. Since the model does not learn adversarial and hard input data during the training phase, some graph data that is perturbed may escape the model's detection in test. 3) The GACL model that introduces contrastive learning is better than GACL-NCL, which proves that contrastive learning is beneficial. 4) Compared to the PHEME dataset, the introduction of adversarial contrastive learning is more effective on Twitter 15 and Twitter 16. It is because that the *rumor* class in PHEME actually includes *unverified*, *ture* and *false*, and the number of *false* rumors only accounts for 7% of the entire dataset. The confusing labels and extreme unbalanced classes conflict with the idea of contrastive learning, so PHEME is more difficult for GACL.

In addition, we studied the impact of the edge perturbation and dropout mask with different rates as shown in Figure 4. Obviously, when the rate of the edge perturbation or the dropout mask is set too large, the model will get poor performance due to the lack of information. Note that the edge perturbation does not change the semantic information of sources and comments, and correct rate setting will help the model achieve better performance. Dropout mask operation will directly cause the lack of text information, so as the mask rate increases, the prediction accuracy becomes lower and lower (in our work, the best mask rate setting is from 5% to 15%).

5.5 Early Rumor Detection

Early rumor detection is also an important way to evaluate models, which aims to detect rumors before they spread widely and cause serious social impact. In this paper, as shown in Figure 5, eight different moments (i.e. 10, 20, ..., 120 minutes) are set to verify whether the model can correctly identify rumors based on the limited information carried at the current early moment.

Figure 5 shows the performance of our GACL model and other different benchmarks in the early rumor detection task. It can be observed that at time 0, when the input data only contains sources, the performance of the models is usually worse. This is due to insufficient training caused by a lack of data, and also a lack of comment information that has been proven to be a key clue to classify rumors. After 10 minutes, the performance of the models has improved significantly, especially the accuracy of the GCN-based models climbing fast and approaching the best performance due to the increasingly rich structural features in the input data. Furthermore, our GACL model is superior and stable at all different moments, and successfully beats other benchmarks, which can prove that the combination of adversarial contrastive learning and graph model can achieve robust and effective detection.

6 CONCLUSION

In this paper, we propose a new rumor detection model named GACL. First, the pre-training model BERT is adopted to obtain the representation of each post in GACL, and then GCN is used to encode the structural information of rumor propagation. Second, contrastive learning is introduced, which can improve the quality of representations by capturing the commonalities between instances of the same class and the differences between instances of different classes. Finally, the AFT module is loaded into the model and trained with the adversarial learning strategy, which aims to generate the adversarial features. These adversarial features are used as hard negative samples in contrastive learning, and also fed into the softmax module as part of the input vector in the training stage, which is beneficial to capture the event-invariant features. Experimental results show that our GACL method is effective and robust for rumor detection on three public real-world datasets, and is significantly superior to other state-of-the-art models in early rumor detection tasks.

Our future work will focus on the fusion and extraction of multimodal information, prejudice detection, and the interpretability of model decisions.

ACKNOWLEDGMENTS

The authors would like to thank the four anonymous reviewers for their comments on this paper. This research was supported by the National Natural Science Foundation of China (Nos. 61836007 and 62006167), and the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD).

REFERENCES

[1] Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 549–556.

[2] Hengyi Cai, Hongshen Chen, Yonghao Song, Zhuoye Ding, Yongjun Bao, Weipeng Yan, and Xiaofang Zhao. 2020. Group-wise contrastive learning for neural dialogue generation. *arXiv preprint arXiv:2009.07543* (2020).

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[4] Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 15750–15758.

[5] Bo Dai and Dahua Lin. 2017. Contrastive learning for image captioning. *arXiv preprint arXiv:1710.02534* (2017).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. *arXiv preprint arXiv:2104.08821* (2021).

[8] Zhiwei Jin, Juan Cao, Han Guo, Yongdong Zhang, and Jiebo Luo. 2017. Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In *Proceedings of the 25th ACM international conference on Multimedia*. 795–816.

[9] Nikhil Ketkar. 2017. Introduction to pytorch. In *Deep learning with python*. Springer, 195–208.

[10] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362* (2020).

[11] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[12] Quanzhi Li, Qiong Zhang, and Luo Si. 2019. eventAI at SemEval-2019 task 7: Rumor detection on social media by exploiting content, user credibility and propagation information. In *Proceedings of the 13th International Workshop on Semantic Evaluation*. 855–859.

[13] Quanzhi Li, Qiong Zhang, and Luo Si. 2019. Rumor detection by exploiting user credibility information, attention and multi-task learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 1173–1179.

[14] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. 2016. Large-margin softmax loss for convolutional neural networks.. In *ICML*, Vol. 2. 7.

[15] Yi-Ju Lu and Cheng-Te Li. 2020. GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. *arXiv preprint arXiv:2004.11648* (2020).

[16] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. (2016).

[17] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. 2015. 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*. 1751–1754.

[18] Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. Association for Computational Linguistics.

[19] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Detect rumor and stance jointly by neural multi-task learning. In *Companion proceedings of the the web conference 2018*. 585–593.

[20] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. Association for Computational Linguistics.

[21] Jing Ma, Wei Gao, and Kam-Fai Wong. 2019. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *The World Wide Web Conference*. 3049–3055.

[22] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. *arXiv preprint arXiv:2005.10200* (2020).

[23] Lahari Poddar, Wynne Hsu, Mong Li Lee, and Shruti Subramaniyam. 2018. Predicting stances in twitter conversations for detecting veracity of rumors: A neural approach. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 65–72.

[24] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. 2020. Gcc: Graph contrastive coding for graph neural network pre-training. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1150–1160.

[25] Kai Shu, Suhang Wang, and Huan Liu. 2017. Exploiting tri-relationship for fake news detection. *arXiv preprint arXiv:1712.07709* 8 (2017).

[26] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification?. In *China National Conference on Chinese Computational Linguistics*. Springer, 194–206.

[27] NGUYEN VAN HA, K Sugiyama, P Nakov, and MY Kan. 2020. FANG: Leveraging Social Context for Fake News Detection Using Graph Representation. (2020).

[28] Penghui Wei, Nan Xu, and Wenji Mao. 2019. Modeling conversation structure and temporal dynamics for jointly predicting rumor stance and veracity. *arXiv preprint arXiv:1909.08211* (2019).

[29] Hanlu Wu, Tengfei Ma, Lingfei Wu, Tariro Manyumwa, and Shouling Ji. 2020. Unsupervised reference-free summary quality evaluation via contrastive learning. *arXiv preprint arXiv:2010.01781* (2020).

[30] Lianwei Wu, Yuan Rao, Yongqiang Zhao, Hao Liang, and Ambreen Nazir. 2020. DTCA: Decision tree-based co-attention networks for explainable claim verification. *arXiv preprint arXiv:2004.13455* (2020).

[31] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer. *arXiv preprint arXiv:2105.11741* (2021).

[32] Xiaoyu Yang, Yuefei Lyu, Tian Tian, Yifei Liu, Yudong Liu, and Xi Zhang. 2020. Rumor Detection on Social Media with Graph Structured Adversarial Learning.. In *IJCAI*. 1417–1423.

[33] Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun. 2021. Decoupled Contrastive Learning. *arXiv preprint arXiv:2110.06848* (2021).

[34] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems* 33 (2020), 5812–5823.

[35] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, Tieniu Tan, et al. 2017. A Convolutional Approach for Misinformation Identification.. In *IJCAI*. 3901–3907.

[36] Chunyuan Yuan, Qianwen Ma, Wei Zhou, Jizhong Han, and Songlin Hu. 2019. Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 796–805.

[37] Zhilu Zhang and Mert R Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *32nd Conference on Neural Information Processing Systems (NeurIPS)*.

[38] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*. 2069–2080.

[39] Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2017. Exploiting context for rumour detection in social media. In *International Conference on Social Informatics*. Springer, 109–123.