Robust Rumor Detection on Social Media with Dynamic Contrastive Learning

Anonymous ACL submission

Abstract

Detecting rumors on social media has become increasingly desirable due to its rapid spread and threat to society and individuals. The propagation-based methods, which have achieved superior detection performance, have recently attracted more and more attention. Existing propagation-based detection models usually only consider the static propagation structure, and ignore the temporal dynamics of propagation structure. The propagation structure also has missing and unreliable interactions. To address these issues, we propose a novel end-to-end Dynamic Contrastive Learning (DCL) model, which integrate dynamic augmented propagation graphs with contrastive learning. Specifically, graph neural networks are adopted to encode the dynamic propagation graphs and dynamic augmented graphs respectively, and the constrastive loss enables the DCL model to learn robust and dynamic representations of messages for rumor detection. Extensive experiments are conducted on two public real-world datasets. Experimental results show that our DCL model significantly improves the effectiveness and robustness compared with the state-of-the-art baselines.

1 Introduction

Social media has become the primary platform for people to access news, express opinions, and engage in social interactions. However, the social media is also a double-edged sword. It has the risk of the proliferation and rapid spread of rumors and poses serious threats to social stability, public trust, and economic order. Therefore, it is urgent to detect rumors effectively on social media.

Traditional detection methods rely on manually designed features and machine learning classifiers to identify rumors (Castillo et al., 2011; Yang et al., 2012; Feng et al., 2012; Kwon et al., 2013). Although these methods are effective to a certain extent, they are limited by the subjectivity and constraints of handcrafted features and fail to capture high-level representations of rumors. In recent years, deep learning models such as Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) have been used to extract high-level representations from the textual contents, user profiles, and propagation structures for rumor detection (Ma et al., 2016, 2018; Liu and Wu, 2018; Li et al., 2019). The propagation structure-based methods, which have achieved superior detection performance, have recently attracted more and more attention. Graph Neural Networks (GNNs) are proposed to model the propagation graph for rumor detection (Bian et al., 2020; Min et al., 2022; Nguyen et al., 2020; Tian et al., 2022). However, these models only consider the static propagation structure of the last state of the rumor, and ignore the temporal dynamics of propagation graphs. The real propagation graph of a social media message is dynamic. Figure 1 illustrates the dynamics of propagation graphs. The spread of social media messages evolves and the dynamics of messages can exhibit the temporal propagation pattern. For example, the different release times of two comments can result in the same spatial structure but different temporal structures.

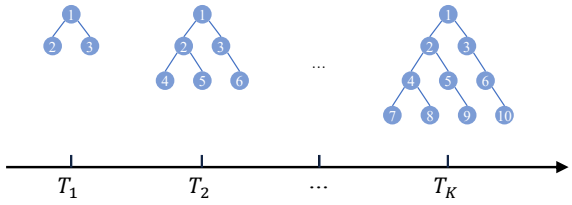


Figure 1: An example of dynamic propagation graph, where each node represents a post, each edge represents the response relationship between nodes, and the dynamic propagation process in the figure is divided into K stages.

Meanwhile, there are some missing and unreliable interactions in the propagation structure during

the actual spread of social media messages (Wei et al., 2022; Lu et al., 2022). The perturbation of the propagation structure can influence the performance of propagation-based methods. Contrastive learning has been proposed to address the issues of the propagation graph and improve the robustness of rumor detection models (You et al., 2020; Chen et al., 2020; Feng et al., 2022). Recent studies (Sun et al., 2022b; Ma et al., 2022, 2023) have adopted supervised or unsupervised contrastive learning for graphs. It is revealed that contrastive learning improves not only the robustness of rumor detection models but also the effectiveness of models. However, few works attempt to research the robustness of models based on temporal dynamics of propagation graphs.

In this paper, we propose a novel end-to-end Dynamic Contrastive Learning (DCL) model, which can integrate the temporal dynamics of propagation graphs with contrastive learning for rumor detection. Specifically, we first divide the propagation process of social media messages into a series of consecutive stages in chronological order by their release time, and the graph of each stage represents a static snapshot of propagation graphs which consists of the source post and the responsive posts until the end of current stage. Then, we use the graph augmentation methods to construct the corresponding augmented graph for each stage and obtain a series of graph snapshots of dynamic propagation graphs and dynamic augmented graphs. Finally, we train a end-to-end model which integrate dynamic augmented graph and contrastive learning to enhance the robustness and effectiveness. The main contributions of this paper can be summarized as follows:

- We propose a novel end-to-end Dynamic Contrastive Learning (DCL) model, which integrates dynamic augmented propagation graphs with contrastive learning to improve the robustness and effectiveness of rumor detection.
- We study a novel problem of the robustness of dynamic propagation graph, and explore graph augmentation methods to improve the robustness of the rumor detection.
- We conduct extensive experiments on two real-world datasets to validate the performance of our DCL model, and the experimental results demonstrate that the robustness and effectiveness of our proposed model are better than those of state-of-the-art baselines.

2 Related Work

2.1 Rumor Detection

Traditional rumor detection methods rely on manually crafted feature engineering to extract rumor features (Castillo et al., 2011; Feng et al., 2012), and these studies depend on the effectiveness of the constructed features. To automatically learn high-level features, recent deep learning-based rumor detection methods have been proposed, mainly categorized into text-based (Ma et al., 2019; Dun et al., 2021; Xu et al., 2022), social context-based (Nguyen et al., 2020; Dou et al., 2021; Tian et al., 2022; Min et al., 2022), and propagation-based approaches (Bian et al., 2020; He et al., 2021; Wei et al., 2021; Sun et al., 2022b; Ma et al., 2022; Liu et al., 2023). The propagation-based rumor detection models have achieved promising performance. Bian et al. (2020) proposed a bidirectional GCN to explore both top-down and bottom-up propagation characteristics of rumors. Wei et al. (2021; 2022) explored uncertainty issues based on propagation structure to capture robust structural features. Methods based on different perturbations and contrastive learning frameworks have also been proposed to obtain robust rumor detection (Sun et al., 2022b; Ma et al., 2022). However, the above methods only consider the static propagation structure of the last state of rumors and neglect the temporal information in the propagation process.

It is crucial to consider dynamic temporal features that capture how rumors evolve on social media. Huang et al. (2020) introduced temporal information to model message propagation. Meanwhile, different approaches to construct dynamic graphs by dividing the rumor propagation process into multiple stages are also proposed to capture the temporal information and network structural information of rumor diffusion (Choi et al., 2021; Song et al., 2022; Sun et al., 2022a). However, these dynamic propagation methods only focus on learning dynamic presentations of the propagation graphs and ignore the robustness of the rumor detection model. In this paper, we focus on the robustness of models based on dynamic propagation graphs and propose a novel end-to-end model for rumor detection.

2.2 Contrastive Learning

In recent years, Contrastive Learning (CL) has attracted widespread attention due to its excellent performance (Zeng and Xie, 2021; Zhu et al., 2021; Suresh et al., 2021; Cui and Jia, 2024; Lee et al.). Contrastive learning explores inherent similarities and differences between different samples (You et al., 2020; Chen et al., 2020; He et al., 2020; Feng et al., 2022). The core idea of contrastive learning is to minimize the distance between positive samples and maximize the distance between negative samples in an embedding space. Contrastive learning methods can be divided into two categories: supervised and unsupervised. While unsupervised contrastive learning does not rely on any label information and only learns the representations by comparing the similarities and differences between samples, supervised contrastive learning utilizes label information to guide contrastive learning. The supervised and unsupervised contrastive learning methods are applied to enhance the model's robustness and effectiveness for rumor detection (Sun et al., 2022b; Ma et al., 2022, 2023). Different from these studies, our work focuses on integrate dynamic augmented graph with contrastive learning in an end-to-end model for rumor detection.

3 Methodology

3.1 Problem Definition

The rumor detection task can be defined as a binary classification problem. Formally, let $C = \{c_1, c_2, \ldots, c_m\}$ be the set of events, where c_i is the i-th event and m is the number of events. For each event $c_i = \{r_i, x_1^i, x_2^i, \ldots, x_{n_i-1}^i, G_i\}$, r_i is the source post and can also be regarded as x_0^i, x_j^i represents the j-th relevant responsive post, n_i is the number of posts in the event c_i. All posts in event c_i are ordered chronologically and the sequence of release time is denoted as $s_i = \{t_0^i, t_1^i, t_2^i, \ldots, t_{n_i-1}^i\}$, where $t_0^i = 0$ represents the release time of the source post and t_j^i represents the release time of the j-th responsive post. $G_i = \langle V_i, E_i \rangle$ is the propagation graph with the root node r_i, where V_i is the set of graph nodes represented with posts and E_i is the set of edges represented with the response relationships among posts. The node feature is denoted as $\mathbf{x}_j^i$, where $\mathbf{x}_0^i$ represents the feature vector of r_i and each other $\mathbf{x}_j^i$ represents the feature vector of x_j^i. The node feature matrix $\mathbf{X}_i \in \mathbb{R}^{n_i \times d}$ is composed by $\mathbf{x}_j^i$, where d is the embedding vector dimension of each node. BERT (Devlin et al., 2018) is used to encode the source post and responsive posts respectively to form the feature matrix $\mathbf{X}_i$. The adjacency matrix is represented as $A_i \in \{0,1\}^{n_i \times n_i}$, where if there is a response relationship between x_u^i and x_v^i, $A_{u,v} = A_{v,u} = 1$, otherwise $A_{u,v} = A_{v,u} = 0$. For simplicity, the subscript i is omitted in the following content.

Rumor detection aims to learn a function $f : C \rightarrow Y$ to classify each event into the categories $Y \in \{F, T\}$ (i.e., Rumor or Non-Rumor).

3.2 Overview

We propose the Dynamic Contrastive Learning (DCL) model. As illustrated in Figure 2, the DCL model primarily consists of three components: (i) a Dynamic Graph Construction Module for building a series of graph snapshots of dynamic propagation graph and dynamic augmented graph respectively; (ii) a Graph Representation Module to obtain the graph representations of two types of dynamic graphs; (iii) a Contrastive Learning Module to learn robust and dynamic representations.

3.3 Dynamic Graph Construction

We construct two kinds of dynamic graphs: one is the dynamic propagation graph, and the other is the dynamic augmented graph.

Dynamic Propagation Graph Construction. To capture the dynamic pattern of the propagation graph, two dynamic graph construction methods have been proposed (Choi et al., 2021). We extend this work and attempt three dynamic graph construction methods: time snapshots, sequential snapshots, and fixed-time snapshots. Time snapshots segment the entire propagation process of an event into multiple stages with equal time intervals, and provide an intuitive imitation of the temporal evolution of rumor dissemination. Sequential snapshots construct each stage based on the number of posts to reflect user engagement intensity. Fixed-time snapshots divide the dissemination into stages with uniform time intervals according to fine-grained time units to emphasize detailed time granularity.

Formally, we divide the propagation graph of an event c into K graph snapshots along the release timeline. Specifically, each graph snapshot has an equal time interval $\Delta t = \frac{t_{n-1} - t_0}{K}$. Based on the source post and relevant responsive posts in each snapshot, we construct a series of graph snapshots of dynamic propagation graph $G = \{G_1, G_2, \ldots, G_K\}$, where $G_k = \langle V_k, E_k \rangle$

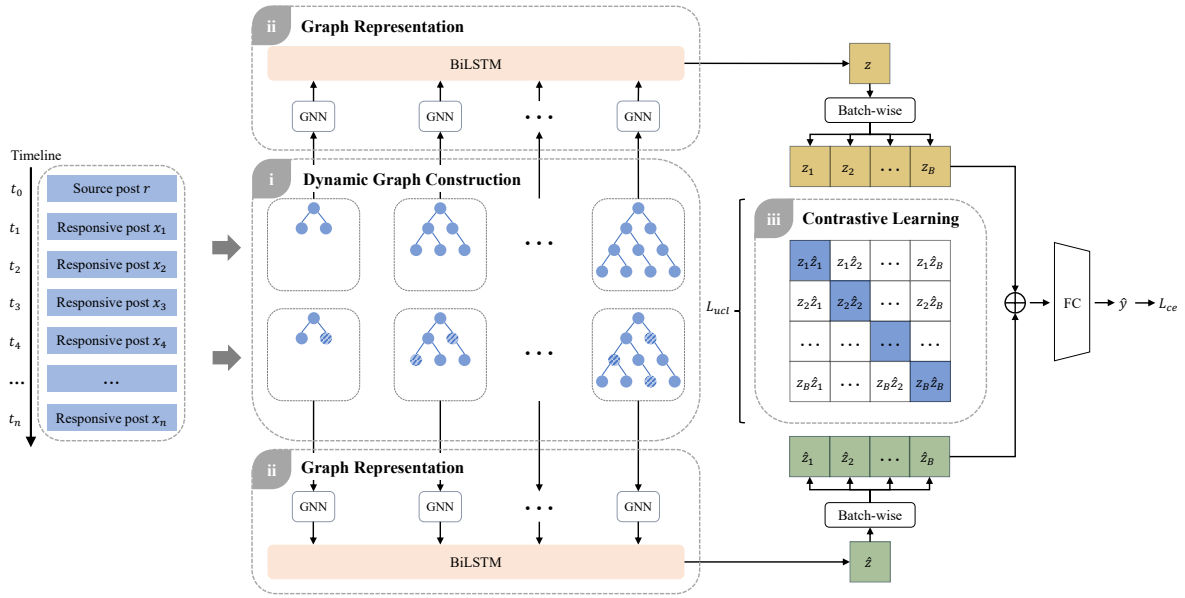


Figure 2: The framework of the proposed DCL model.

is the k-th graph snapshot. $V_k = \{c_k | s_k \leq k\Delta t\}$ is the set of vertices in G_k. E_k represents the set of edges in G_k. The adjacency matrix is $A_k \in \{0,1\}^{n_k \times n_k}$, and the node feature matrix denotes as $\mathbf{X}_k \in \mathbb{R}^{n_k \times d}$, where d is the embedding vector dimension of each node.

Dynamic Augmented Graph Construction. The propagation-based rumor detection models usually suffer from missing and unreliable interactions in the propagation structure. Graph augmentation methods can alleviate this problem and enhance the robustness of models by modifying node or edge attributes during training. We explore three methods of graph augmentation: node mask, edge drop, and feature mask. Node mask randomly masks node features during training and enables the model to learn robust representations in the absence of certain node information. Edge drop randomly removes connections of some edges during training, and it is used to allow the model to robustly predict results when partial connection information is lost. Feature mask randomly masks node dimensional features during training, enabling the model to adapt and generalize in case of incomplete feature information. The experimental results demonstrate that the node mask outperforms the others. Therefore, we apply the node mask strategy to perform the graph augmentation in DCL.

Specifically, given a graph snapshot G_k of dynamic propagation graph, we randomly mask some nodes in the snapshot to obtain the augmented graph snapshot $\hat{G}_k$ at each training epoch. Formally, given node feature matrix $\mathbf{X}_k \in \mathbb{R}^{n_k \times d}$ and the masking rate r_{mask}, the node feature matrix $\hat{\mathbf{X}}_k \in \mathbb{R}^{n_k \times d}$ of $\hat{G}_k$ is computed as follows:

$$\hat{\mathbf{X}}_k = \mathbf{M} \odot \mathbf{X}_k \tag{1}$$

where $\mathbf{M} \in \{0,1\}^{n_k \times d}$ is the masking matrix by randomly generating $n_k \times r_{mask}$ rows of zero vector and $\odot$ denotes the element-wise multiplication. Note that the masked node does not include the root node. The masked ones in the previous graph snapshot will also be masked in the next graph snapshot.

After applying the node mask operation on each graph snapshot of dynamic propagation graph, we obtain a series of graph snapshots of dynamic augmented graph $\hat{G} = \{\hat{G}_1, \hat{G}_2, \ldots, \hat{G}_K\}$, where $\hat{G}_k = \langle V_k, E_k \rangle$ is the k-th augmented graph snapshot. The adjacency matrix is $A_k \in \{0,1\}^{n_k \times n_k}$ and the node feature matrix is $\hat{\mathbf{X}}_k \in \mathbb{R}^{n_k \times d}$.

3.4 Graph Representation

We encode a series of graph snapshots of dynamic propagation graph and dynamic augmented graph and obtain their graph representations.

The Graph Isomorphism Network (GIN) (Xu et al., 2019) is used as the Graph Neural Network (GNN) in our proposed DCL model to capture the representations of the graph snapshots of dynamic propagation graph $G = \{G_1, G_2, \ldots, G_K\}$ and the graph snapshots of dynamic augmented graph $\hat{G} = \{\hat{G}_1, \hat{G}_2, \ldots, \hat{G}_K\}$.

Given a graph snapshot G_k, at the l-th layer, the GIN encoder updates the hidden feature vector $h_v^{(l)}$ for node v using the following aggregation function:

$$h_v^{(l)} = \text{MLP}\left((1+\epsilon^{(l)}) \cdot h_v^{(l-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(l-1)}\right) \tag{2}$$

where ϵ is a learnable parameter, $\mathcal{N}(v)$ is the set of neighboring nodes of node v, and $h_v^{(0)} = \mathbf{x}_v$ is the initial feature vector of node v. This process is iterated for all nodes until the l-th layer.

A two-layer GIN is employed. After encoding steps on the graph snapshots G_k and $\hat{G}_k$, we can obtain the representations H_k and $\hat{H}_k$. Then mean-pooling operator is used to aggregate the information to obtain graph representations $\mathbf{h}_k$ and $\hat{\mathbf{h}}_k$.

$$H_k = \text{GIN}(A_k, \mathbf{X}_k) \tag{3}$$

$$\mathbf{h}_k = \text{MEAN}(H_k) \tag{4}$$

$$\hat{H}_k = \text{GIN}(A_k, \hat{\mathbf{X}}_k) \tag{5}$$

$$\hat{\mathbf{h}}_k = \text{MEAN}(\hat{H}_k) \tag{6}$$

Finally, the dynamic propagation graph is represented as $\mathbf{h}$. Similarly, the dynamic augmented graph $\hat{\mathbf{h}}$ is obtained in the same manner.

$$\mathbf{h} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_K\} \tag{7}$$

$$\hat{\mathbf{h}} = \{\hat{\mathbf{h}}_1, \hat{\mathbf{h}}_2, \ldots, \hat{\mathbf{h}}_K\} \tag{8}$$

After obtaining the graph representations of the dynamic propagation graph and the dynamic augmented graph, we adopt Bidirectional Long Short-Term Memory (BiLSTM) (Hochreiter and Schmidhuber, 1997) to model graph representation sequences in both forward and backward directions. BiLSTM can capture long-term dependencies effectively and identify latent patterns in dynamic graphs. The above process can be described formally as follows:

$$\begin{aligned} \overrightarrow{z} &= \overrightarrow{\text{LSTM}}(\mathbf{h}) \\ \overleftarrow{z} &= \overleftarrow{\text{LSTM}}(\mathbf{h}) \end{aligned} \tag{9}$$

$$\begin{aligned} \overrightarrow{\hat{z}} &= \overrightarrow{\text{LSTM}}(\hat{\mathbf{h}}) \\ \overleftarrow{\hat{z}} &= \overleftarrow{\text{LSTM}}(\hat{\mathbf{h}}) \end{aligned} \tag{10}$$

Then we concatenate the forward state and the backward state to obtain the graph representations encoded by BiLSTM, i.e., z and $\hat{z}$, where CONCAT represents the concatenate operation.

$$\begin{aligned} z &= \text{CONCAT}(\overrightarrow{z}, \overleftarrow{z}) \\ \hat{z} &= \text{CONCAT}(\overrightarrow{\hat{z}}, \overleftarrow{\hat{z}}) \end{aligned} \tag{11}$$

Finally, we obtain the dynamic propagation graph representation $Z = \{z_1, z_2, \ldots, z_{\mathcal{B}}\}$ and the dynamic augmented graph representation $\hat{Z} = \{\hat{z}_1, \hat{z}_2, \ldots, \hat{z}_{\mathcal{B}}\}$, where $\mathcal{B}$ refers to the batch size, not the total number of events, since the model is trained in a batch-wise manner for the practical implementation.

3.5 Contrastive Learning

We formulate a loss function as the optimization objective for unsupervised contrastive learning, aiming to maximize consistency between the graph representations of the dynamic propagation graph and its corresponding dynamic augmented graph. Specifically, given the dynamic propagation graph representation Z and the dynamic augmented graph representation $\hat{Z}$, we can construct $\mathcal{B}^2$ representation pairs $\{(z_i, \hat{z}_j) | i, j = 1, \ldots, \mathcal{B}\}$. Among them, only the $\mathcal{B}$ pairs with $i = j$ are correct matches of the dynamic propagation graph representations and the corresponding dynamic augmented graph representations, which are defined as positive samples. While the remaining $\mathcal{B}^2 - \mathcal{B}$ pairs are wrong matches and are regarded as negative samples. Our contrastive learning aims to maximize the similarity of the $\mathcal{B}$ correct matches while minimizing the similarity of the $\mathcal{B}^2 - \mathcal{B}$ wrong matches. To compute the similarity for the $\mathcal{B}^2$ pairs, we construct the similarity matrix $\Psi \in \mathbb{R}^{\mathcal{B} \times \mathcal{B}}$ to capture the pairwise similarities as follows:

$$\Psi = exp(\frac{sim(Z, \hat{Z})}{\tau}) \tag{12}$$

where $sim(\cdot)$ denotes the cosine similarity function such as $sim(Z, \hat{Z}) = \frac{Z\hat{Z}^{\mathrm{T}}}{||Z|| \cdot ||\hat{Z}||}$, τ is a temperature parameter to scale the similarity values.

To formulate the contrastive loss based on the graph representations, we apply the multi-class N-pair loss (Sohn, 2016) by considering the row-wise cross entropy loss with respect to the row index.

$$\mathcal{L}_{ucl} = \text{CE}(\Psi, \text{I}) \tag{13}$$

where $\text{I} \in \mathbb{R}^{\mathcal{B} \times \mathcal{B}}$ represents the identity matrix for contrastive training, and CE denotes the cross entropy loss applied row-wise to the input matrix Ψ.

4 Training Objective

We obtain the dynamic propagation graph representation Z and the dynamic augmented graph representation $\hat{Z}$. Then we concatenate them to integrate the information as $\mathrm{Z} = \mathrm{CONCAT}(Z, \hat{Z})$. Then Z is fed into fully connection layers and a softmax layer, and the output is calculated as

$$\hat{y} = softmax(W_f \mathrm{Z} + b_f) \tag{14}$$

where $\hat{y} \in \mathbb{R}^{\mathcal{B} \times \mathcal{C}}$ is the predicted probability distribution. W_f and b_f are weight and bias parameters respectively.

$$\mathcal{L}_{ce} = -\frac{1}{\mathcal{B}} \sum_{b=1}^{\mathcal{B}} \sum_{c=1}^{\mathcal{C}} y_{b,c} log(\hat{y}_{b,c}) \tag{15}$$

where $y_{b,c}$ denotes ground-truth label and $\hat{y}_{b,c} \in \mathbb{R}^{1 \times \mathcal{C}}$ denotes the predicted probability distribution of index $b \in \{1, \ldots, \mathcal{B}\}$ belongs to class $c \in \{1, \ldots, \mathcal{C}\}$. In our binary classification task, $\mathcal{C} = 2$ denotes the number of classes.

Our training object contains a unsupervised contrastive loss $\mathcal{L}_{ucl}$ and a cross-entropy loss $\mathcal{L}_{ce}$. Finally, our proposed end-to-end model can be trained by minimizing the following loss:

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{ucl} \tag{16}$$

where λ is the trade-off parameter.

5 Experiments

We evaluate our DCL model on RumorEval (Derczynski et al., 2017) and TWITTER (Ma et al., 2017). Refer to Appendix A.1, Appendix A.2 and Appendix A.3 for details of the datasets, comparison models and experimental setup respectively.

5.1 Overall Performance

Table 1 shows the results of rumor detection on RumorEval and TWITTER datasets. Our model DCL outperforms all the baseline models in terms of all metrics, which confirms the effectiveness of temporal dynamics of propagation graphs and contrastive learning. Propagation-based models BiGCN and EBGCN have the worst performance. The possible reason is that these models only capture spatial information, which makes them vulnerable to noise and heavily depends on large amounts of labeled data. Models (i.e., GACL, RDEA, and TrustRD) that integrate propagation structures with contrastive learning, achieve better results than those based solely on propagation structures. GACL utilizes supervised contrastive learning, while RDEA and TrustRD adopt self-supervised contrastive learning. GACL, RDEA and TrustRD capture information from static propagation graphs. The dynamic propagation-based model DynGCN captures both the temporal and spatial information. However, DynGCN fails to effectively learn robust representations of dynamic propagation graphs. In contrast, our model emphasizes the temporal dynamics of propagation graphs and learns robust representations with contrastive learning. Graph augmentation strategies are used to enrich graph data. By using dynamic propagation graphs and dynamic augmented graphs simultaneously, our model enhances the ability to capture both spatial and temporal information. Additionally, contrastive learning enables our model to obtain high-quality feature representations and achieves superior performance.

5.2 Ablation Study

To analyze the effect of different modules in our proposed model DCL, we compare it with the following variants: (1) w/o Prop: removing dynamic propagation graphs; (2) w/o Aug: deleting dynamic augmented graphs; (3) w/o CL: neglecting contrastive learning loss; (4) w/o Dyn: using static graphs instead of dynamic graphs. Specifically, we only use the last graph snapshots of the two types of dynamic graphs.

The experimental results are shown in Table 2. F1-score of Rumor (F) category is used as one of evaluation metrics. Because the detection performance drops when any one of these components is removed, each component plays an independent role in detecting rumors on social media. Specifically, when the dynamic propagation graphs are removed, the accuracy on the RumorEval dataset and TWITTER dataset drops by 1.04% and 0.85%. It validates the significance of spatial structure information. If the dynamic augmented graphs are removed, the accuracy on two datasets drops by 0.67% and 1.18%. Thus, augmented graphs can enhance the effectiveness of the model. When contrastive learning is removed, the accuracy on two datasets drops by 0.71% and 0.58%. So, contrastive learning enables our model to effectively capture the intrinsic relationships within data features and obtain high-quality representations. When dynamic graphs are replaced with static graphs, the accuracy on two datasets drops by 2.71% and 1.10%. The temporal information of dynamic graphs is impor-

Method	Class	RumorEval				TWITTER			
		Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
Bi-GCN	F	0.7115	0.7314	0.7599	0.7454	0.7591	0.7595	0.7419	0.7506
	T		0.7042	0.6562	0.6794		0.7771	0.7732	0.7751
EBGCN	F	0.6952	0.7100	0.7652	0.7366	0.7539	0.7449	0.7428	0.7438
	T		0.6812	0.6125	0.6450		0.7657	0.7640	0.7648
GACL	F	0.7250	0.7974	0.7385	0.7668	0.7609	0.7987	0.6781	0.7335
	T		0.7591	0.7091	0.7332		0.7454	0.8377	0.7889
RDEA	F	0.7321	0.7513	0.7890	0.7697	0.7855	0.7942	0.7496	0.7713
	T		0.7434	0.6648	0.7019		0.7857	0.8181	0.8016
TrustRD	F	0.7267	0.7298	0.8175	0.7712	0.7695	0.7751	0.7359	0.7550
	T		0.7617	0.6195	0.6833		0.7749	0.7974	0.7860
DynGCN	F	0.7377	0.7471	0.7617	0.7543	0.7693	0.7647	0.7495	0.7570
	T		0.7092	0.6823	0.6955		0.7759	0.7893	0.7825
DCL	F	**0.8121**	**0.8413**	**0.8215**	**0.8313**	**0.8308**	**0.8368**	**0.8034**	**0.8198**
	T		**0.7991**	**0.8009**	**0.8000**		**0.8319**	**0.8554**	**0.8435**

Table 1: Rumor detection results on RumorEval and TWITTER datasets. Abbrev.: Rumor (F), Non-Rumor (T).

Model	RumorEval		TWITTER	
	Acc.	F1	Acc.	F1
DCL	**0.8121**	**0.8313**	**0.8308**	**0.8198**
w/o Prop	0.8017	0.8181	0.8223	0.8086
w/o Aug	0.8054	0.8272	0.8190	0.8034
w/o CL	0.8050	0.8229	0.8250	0.8091
w/o Dyn	0.7850	0.8071	0.8198	0.8100

Table 2: Results of ablation study on two datasets.

Model	RumorEval		TWITTER	
	Acc.	F1	Acc.	F1
temporal	**0.8121**	**0.8313**	**0.8308**	**0.8198**
sequential	0.8042	0.8218	0.8245	0.8095
fixed-time	0.8075	0.8265	0.8219	0.8057

Table 3: Results of different dynamic graph construction on two datasets.

tant to improve performance.

5.3 Different Dynamic Graph Construction

To explore different methods of dynamic graph construction, we follow (Choi et al., 2021; Song et al., 2022) and compare sequential snapshots and fixed-time snapshots with time snapshots used in the DCL model. In the sequential snapshots method, the dynamic graphs for each event consist of K snapshots based on the number of posts ($K = 3$). In the fixed-time snapshots method, each event contains multiple snapshots which are fixed in the specified interval $[0.0, 0.5, 1.0, 1.5, 2.0, 4.0, 8.0, max]$ hours ($K = 8$).

From Table 3, the time snapshots method used in the DCL model achieves the best performance. F1-score of Rumor (F) category is used as one of evaluation metrics. Time snapshots effectively capture the temporal evolution patterns during the propagation, and understand propagation behaviors across different time points. In contrast, sequential snapshots are unstable due to drastic fluctuations in the number of posts within certain periods. This method focuses solely on the number of posts and overlooks the temporal dynamics and potentially temporal features. Fixed-time snapshots fail to imitate the actual propagation patterns, resulting in either insufficient or redundant information. When the propagation characteristics of rumors are not prominent within fixed intervals, this method may be limited in its ability to accurately detect rumors. The experimental results verify the superiority of time snapshots in our model for rumor detection.

5.4 Different Graph Augmentation Methods

To investigate the impact of different graph augmentation methods on model performance, we compare three methods: node mask, edge drop, and feature mask. F1-score of Rumor (F) category is used as one of evaluation metrics. The experimental results are shown in Table 4. Among these augmentation methods, the node mask used in DCL achieves the best effectiveness. Node mask improves the effectiveness of the proposed model and enhances robustness by enabling learning even

Model	RumorEval		TWITTER	
	Acc.	F1	Acc.	F1
node mask	**0.8121**	**0.8313**	**0.8308**	**0.8198**
edge drop	0.8025	0.8176	0.8244	0.8128
feature mask	0.8004	0.8196	0.8092	0.7899

Table 4: Results of different augmentation on two datasets.

with missing node information. Edge drop leads to the instability of the graph structure and affects the model's performance.Because nodes are crucial for graph propagation, the feature mask will lose detailed information and influence classification performance. The experimental results demonstrate that the node mask method is better than others in improving the effectiveness of the DCL model.

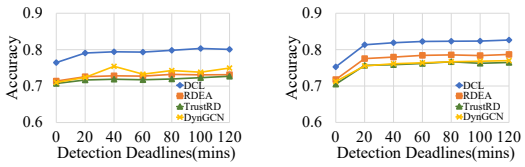
5.5 Robustness Study

We conduct *early rumor detection* and *incremental masking rates* experiments to evaluate the robustness of the DCL model.

Early Rumor Detection. This Experiment aims to detect rumors on social media in an early stage, which is beneficial for real-time rumor detection. To construct the early detection task, we follow (Sun et al., 2022b) and set up a series of detection deadlines regarding the time elapsed since the source post is released.

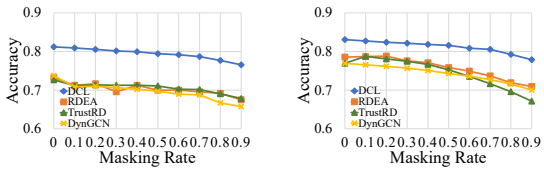
Figure 3 shows the performance of DCL in early rumor detection compared with RDEA, TrustRD, and DynGCN at various deadlines on the RumorEval and TWITTER datasets. It can be observed that at time 0, all models exhibit poor performance due to limited training data from insufficient responsive posts. Subsequently, all models show improvement in accuracy as the detection deadlines elapse. Because the DCL model consistently achieves higher accuracy than other models at each deadline, the DCL model has the best robustness in early rumor detection.

Incremental Masking Rates. To investigate the robustness of DCL, we apply the node mask to the test sets. The masking rate is incrementally increased within $[0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]$. We observe the accuracy of DCL, RDEA, TrustRD, and DynGCN. As illustrated in Figure 4a, on the RumorEval dataset, the accuracy of various models generally remains high at lower masking rates. When the masking rate increases, the accuracy of RDEA and TrustRD gradually decreases slower



(a) RumorEval dataset (b) TWITTER dataset

Figure 3: Results of early rumor detection on two datasets.



(a) RumorEval dataset (b) TWITTER dataset

Figure 4: Results of incremental masking rates on two datasets.

than DynGCN. Figure 4b shows that on the TWITTER dataset, RDEA and TrustRD have an initial increase in accuracy, but then sharply decline. The increase in masking rate leads to a gradual decline in accuracy for all models, which can reflect the level of robustness. In contrast, the accuracy of DCL decreases more slowly and consistently outperforms the comparison models. Moreover, the change in DCL's accuracy is significantly smaller than that of the comparison models, maintaining high accuracy even at higher masking rates. The experimental results demonstrate that our model is robust and can effectively alleviate the effect of missing data.

6 Conclusion

In this paper, we propose a novel end-to-end Dynamic Contrastive Learning (DCL) model for rumor detection. Dynamic propagation graphs and dynamic augmented graphs are fed into our proposed model and Graph Isomorphism Network (GIN) and BiLSTM are used to capture representations of these graphs. Moreover, we train the model to integrate dynamic augmented graphs and contrastive learning. Experiments on two public datasets demonstrate that DCL model outperforms the state-of-the-art baselines in both robustness and effectiveness. In the future, we will explore more dynamic features to enhance the performance of rumor detection further.

7 Limitations

Our model only considers implicit time granularity, and ignore explicit timestamp information. The time snapshots are also relatively coarse-grained. In addition, although the graph augmentation method and contrastive learning can improve the effectiveness and robustness of rumor detection, our model does not consider more contrastive learning methods.

References

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*, volume 34, pages 549–556.

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684.

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*, pages 1597–1607. PMLR.

Jiho Choi, Taewook Ko, Younhyuk Choi, Hyungho Byun, and Chong-kwon Kim. 2021. Dynamic graph convolutional networks with attention mechanism for rumor detection on social media. *Plos one*, 16(8):e0256039.

Chaoqun Cui and Caiyan Jia. 2024. Propagation tree is not deep: Adaptive graph contrastive learning approach for rumor detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 73–81.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. SemEval-2017 task 8: RumourEval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76, Vancouver, Canada. Association for Computational Linguistics.

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*.

Yingtong Dou, Kai Shu, Congying Xia, Philip S Yu, and Lichao Sun. 2021. User preference-aware fake news detection. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 2051–2055.

Yaqian Dun, Kefei Tu, Chen Chen, Chunyan Hou, and Xiaojie Yuan. 2021. Kan: Knowledge-aware attention network for fake news detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 81–89.

Shengyu Feng, Baoyu Jing, Yada Zhu, and Hanghang Tong. 2022. Adversarial graph contrastive learning with information regularization. In *Proceedings of the ACM Web Conference 2022*, pages 1362–1371.

Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 171–175.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Zhenyu He, Ce Li, Fan Zhou, and Yi Yang. 2021. Rumor detection on social media with event augmentations. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, pages 2020–2024.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Qi Huang, Chuan Zhou, Jia Wu, Luchen Liu, and Bin Wang. 2020. Deep spatial–temporal structure learning for rumor detection on twitter. *Neural Computing and Applications*, 35:12995–13005.

Nikhil Ketkar, Jojo Moolayil, Nikhil Ketkar, and Jojo Moolayil. 2021. Introduction to pytorch. *Deep learning with python: learn best practices of deep learning models with PyTorch*, pages 27–91.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. 2013. Prominent features of rumor propagation in online social media. In *2013 IEEE 13th international conference on data mining*, pages 1103–1108. IEEE.

Seanie Lee, Dong Bok Lee, and Sung Ju Hwang. Contrastive learning with adversarial perturbations for conditional text generation. In *International Conference on Learning Representations*.

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*, pages 1173–1179.

Leyuan Liu, Junyi Chen, Zhangtao Cheng, Wenxin Tai, and Fan Zhou. 2023. Towards trustworthy rumor detection with interpretable graph structural learning. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 4089–4093.

Yang Liu and Yi-Fang Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Menglong Lu, Zhen Huang, Binyang Li, Yunxiang Zhao, Zheng Qin, and DongSheng Li. 2022. Sifter: A framework for robust rumor detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:429–442.

Guanghui Ma, Chunming Hu, Ling Ge, Junfan Chen, Hong Zhang, and Richong Zhang. 2022. Towards robust false information detection on social networks with contrastive learning. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1441–1450.

Jiachen Ma, Jing Dai, Yong Liu, Meng Han, and Chunyu Ai. 2023. Contrastive learning for rumor detection via fitting beta mixture model. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 4160–4164.

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.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 708–717, Vancouver, Canada. Association for Computational Linguistics.

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

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*, pages 3049–3055.

Erxue Min, Yu Rong, Yatao Bian, Tingyang Xu, Peilin Zhao, Junzhou Huang, and Sophia Ananiadou. 2022. Divide-and-conquer: Post-user interaction network for fake news detection on social media. In *Proceedings of the ACM web conference 2022*, pages 1148–1158.

Van-Hoang Nguyen, Kazunari Sugiyama, Preslav Nakov, and Min-Yen Kan. 2020. Fang: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1165–1174.

Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29.

Chenguang Song, Yiyang Teng, Yangfu Zhu, Siqi Wei, and Bin Wu. 2022. Dynamic graph neural network for fake news detection. *Neurocomputing*, 505:362–374.

Mengzhu Sun, Xi Zhang, Jiaqi Zheng, and Guixiang Ma. 2022a. Ddgcn: Dual dynamic graph convolutional networks for rumor detection on social media. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4611–4619.

Tiening Sun, Zhong Qian, Sujun Dong, Peifeng Li, and Qiaoming Zhu. 2022b. Rumor detection on social media with graph adversarial contrastive learning. In *Proceedings of the ACM Web Conference 2022*, pages 2789–2797.

Susheel Suresh, Pan Li, Cong Hao, and Jennifer Neville. 2021. Adversarial graph augmentation to improve graph contrastive learning. *Advances in Neural Information Processing Systems*, 34:15920–15933.

Lin Tian, Xiuzhen Jenny Zhang, and Jey Han Lau. 2022. Duck: Rumour detection on social media by modelling user and comment propagation networks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4939–4949.

Lingwei Wei, Dou Hu, Wei Zhou, and Songlin Hu. 2022. Uncertainty-aware propagation structure reconstruction for fake news detection. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2759–2768.

Lingwei Wei, Dou Hu, Wei Zhou, Zhaojuan Yue, and Songlin Hu. 2021. Towards propagation uncertainty: Edge-enhanced Bayesian graph convolutional networks for rumor detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3845–3854, Online. Association for Computational Linguistics.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How powerful are graph neural networks? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Weizhi Xu, Junfei Wu, Qiang Liu, Shu Wu, and Liang Wang. 2022. Evidence-aware fake news detection with graph neural networks. In *Proceedings of the ACM web conference 2022*, pages 2501–2510.

Fan Yang, Yang Liu, Xiaohui Yu, and Min Yang. 2012. Automatic detection of rumor on sina weibo. In *Proceedings of the ACM SIGKDD workshop on mining data semantics*, pages 1–7.

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*, 33:5812–5823.

Jiaqi Zeng and Pengtao Xie. 2021. Contrastive self-supervised learning for graph classification. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 35, pages 10824–10832.

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*, pages 2069–2080.

A Appendix

A.1 Dataset

We conduct experiments on two public real-world datasets: RumorEval (Derczynski et al., 2017) and TWITTER (Ma et al., 2017), which are collected from the popular social media platform Twitter. Both datasets contain post text, timestamp, and propagation information. The datasets have two binary labels: Rumor (F) and Non-Rumor (T). We retain events with more than 3 comments for each dataset in our experiments. The statistics of the two datasets are shown in Table 5.

A.2 Comparison Models

We compare the proposed model with the following baselines:

- **Bi-GCN** (Bian et al., 2020): A rumor detection model that utilizes a Bi-directional Graph Convolutional Network (GCNs) to model the top-down rumor propagation and the bottom-up rumor dispersion.
- **EBGCN** (Wei et al., 2021): A Bayesian GCNs-based rumor detection model that models uncertainty within rumor propagation.
- **GACL** (Sun et al., 2022b): A GCNs-based rumor detection model using adversarial and contrastive learning.
- **RDEA** (He et al., 2021): A rumor detection model that incorporates a self-supervised learning module, which uses graph augmentation strategies and contrastive learning.
- **TrustRD** (Liu et al., 2023): An interpretable rumor detection model which combines a self-supervised learning module and Bayesian networks.
- **DynGCN** (Choi et al., 2021): A dynamic rumor detection model that utilizes GCNs and attention mechanisms to capture both structural and temporal information.

A.3 Experimental Setup

The proposed model is implemented by PyTorch (Ketkar et al., 2021). Following (Bian et al., 2020), we randomly split the datasets into five equal parts, and conduct the 5-fold cross-validation to obtain final results. On two datasets, we evaluate the Accuracy (Acc.) over the two categories, and Precision (Prec.), Recall (Rec.), F1-score (F1) on each class.

Statistics	RumorEval	TWITTER
# Events	245	1077
# Posts	4145	60207
# Non-Rumors	112	564
# Rumors	133	513
# Avg. time length	12 Hours	416 Hours

Table 5: Statistics of the datasets. “#" is “the number of".

The parameters are optimized by Adam algorithm (Kingma and Ba, 2014). The dimension of each node's hidden vectors is 128. The masking rate in the node mask is set to 0.2. The temperature parameter τ is set to 0.8 and 0.4 for the RumorEval and TWITTER datasets. The trade-off parameter λ is set to 0.1. The number of graph snapshots K for each event is set to 3.