FGDGNN: Fine-Grained Dynamic Graph Neural Network for Rumor Detection on Social Media

Anonymous ACL submission

Abstract

Detecting rumors on social media has become a crucial issue. Propagation structure-based methods have recently attracted increasing attention. When the propagation structure is represented by the dynamic graph, temporal information is considered. However, existing rumor detection models using dynamic graph typically focus only on coarse-grained temporal information and ignore the fine-grained temporal dynamics within individual snapshots and across snapshots. In this paper, we propose a novel Fine-Grained Dynamic Graph Neural Network (FGDGNN) model, which can incorporate the fine-grained temporal information of dynamic propagation graph in the intra-snapshot and dynamic embedding update mechanism in the inter-snapshots into a unified framework for rumor detection. Specifically, we firstly construct the edge-weighed propagation graph and the edge-aware graph isomorphism network is proposed. To get fine-grained temporal representations across snapshots, we propose an embedding transformation layer to update node embedding. Finally, we integrate the temporal information in the inter-snapshots at the graph level to enhance the effectiveness of the proposed model. Extensive experiments conducted on three public real-world datasets demonstrate that our FGDGNN model achieves significant improvements compared with the state-of-the-art baselines.

1 Introduction

Social media has become the major platform for information sharing among the public. However, the widespread of social media also brings significant challenges. One of the main challenges is the rapid spread of rumor, which can pose severe risks to public trust, people's health and social stability. Therefore, it has become increasingly important to develop effective methods for identifying and combating rumors.

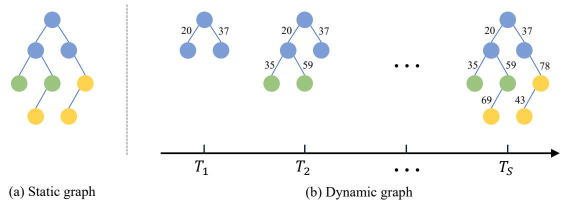


Figure 1: An example of event propagation graph on social media. (a) Static graph. Each node represents a post and each edge represents the response relationship without temporal information. (b) Dynamic graph. Each node represents a post and each edge represents the response relationship with an associated temporal information. The dynamic propagation process in the example is divided into S snapshots.

Previous researches 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). To overcome the limitations of handcrafted features, deep learning models such as Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) have been used to extract high-level representations automatically from the content-based methods and propagation structure-based methods 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 attracted more and more attention in recent years (Bian et al., 2020; Min et al., 2022; Nguyen et al., 2020). However, existing propagation structure-based methods usually consider the static graph structure of the final state of rumor propagation, and ignore the temporal dynamics of the rumor propagation. Figure 1(a) illustrates a static graph structure of rumor propagation. The temporal features of propagation refer to the order and interval of the replied or retweeted posts along the timeline, which records the timestamps of user engagements. Therefore, some studies (Lao et al., 2021; Chang et al., 2024; Choi et al., 2021; Song et al., 2021; Sun et al., 2022a; Xu et al., 2024) have explored the temporal dynamics of news events and proposed the dynamic graphs to model the spread of rumors on social media. A dynamic graph of event propagation is shown in Figure 1(b). Those methods, which usually are built by Graph Neural Network (GNN), emphasize the transformation and aggregation of graph features, but fail to capture the detailed temporal features of propagation, such as the speed, depth and breadth that indicate the propagation effect of rumor event.

To address this issue, the dynamic graph of propagation including a series of snapshots is constructed to model temporal dynamics (Choi et al., 2021; Song et al., 2021; Sun et al., 2022a; Xu et al., 2024). Those works treat snapshots of dynamic graphs as isolated from one another or only allow for coarse-grained interactions between two consecutive snapshots (i.e. in the inter-snapshots). Fine-grained temporal features are required to capture the details of propagation. We divide the temporal granularity of propagation into the node-level and edge-aware granularity respectively. The edge-weighted propagation graphs, which is enabled to represent the speed, depth and breadth of propagation, is used to describe edge-aware granularity in the intra-snapshot. As shown in Figure 1(b), the weighted graph is used to represent propagation graph and the weight on the edge indicates the time interval between when a post is created and when its responded post is created. The node-level granularity is adopted to capture the temporal dynamics in the inter-snapshots.

In this paper, we propose a novel Fine-Grained Dynamic Graph Neural Network (FGDGNN) model, which can incorporate the edge-aware temporal information of dynamic propagation graph in the intra-snapshot and the node-level dynamic update in the inter-snapshots into a unified framework for rumor detection. Specifically, we firstly construct the edge-weighed propagation graph in which the time intervals are used as the edge weights. The propagation process is represented as a sequence of graph snapshots. Then, Edge-Aware Graph Isomorphism Network (EAGIN) is proposed to make full use of the edge weights to capture detailed temporal features in the intra-snapshot. To get fine-grained temporal representations in the inter-snapshots, we propose an embedding transformation layer to update node embeddings. Finally, we integrate the temporal information in the inter-snapshot at the graph level with the framework to enhance the effectiveness of the proposed FGDGNN model.

The main contributions of this paper can be summarized as follows:

- We propose a novel Fine-Grained Dynamic Graph Neural Network (FGDGNN) model, which integrates the edge-aware temporal information and the node-level dynamic update mechanism in the dynamic propagation graph.
- We propose a method of constructing an edge-weighted graph which is enabled to represented fine-grained temporal features, and study a novel problem of temporal granularity of dynamic propagation graph to explore temporal information in the intra-snapshot and inter-snapshots for rumor detection.
- We conduct extensive experiments on three real-world datasets to demonstrate the effectiveness of our proposed model on rumor detection.

2 Related Work

2.1 Rumor Detection

Early rumor detection methods primarily rely on hand-crafted feature engineering and statistical machine learning methods to extract features (Castillo et al., 2011; Yang et al., 2012; Feng et al., 2012; Kwon et al., 2013). Recently, deep learning models have been proposed for rumor detection, including content-based (Ma et al., 2019; Nguyen et al., 2020; Dun et al., 2021; Xu et al., 2022; Min et al., 2022) and propagation structure-based methods (He et al., 2021; Wei et al., 2021; Ma et al., 2022). Propagation structure-based models aim to capture structural characteristics to enhance rumor detection. With the growing attention, a variety of models leveraging propagation structures have been widely explored. Some works (Bian et al., 2020; Wei et al., 2021; Tao et al., 2024) develop the propagation graph from both top-down and bottom-up perspectives to capture rumor propagation nature. With the application of augmentation techniques and contrastive learning, (Sun et al., 2022b; He et al., 2021; Liu et al., 2023; Cui and Jia, 2024) build rumor detection models to improve the understanding of the propagation process. In addition, the development of static graph approaches for rumor detection have also provided valuable insights for dynamic graphs.

Works like (Lao et al., 2021; Chang et al., 2024) integrate temporal information into node features to model the evolving nature of rumor propagation. Meanwhile, methods such as (Choi et al., 2021; Song et al., 2021; Sun et al., 2022a; Xu et al., 2024) treat the dynamic propagation process by dividing the graph into temporal snapshots, simulating how rumors spread over time. However, these dynamic propagation methods only focus on the coarse-grained temporal information. They fail to effectively capture the fine-grained temporal information. Meanwhile, existing methods either treat snapshots of dynamic graphs as isolated from one another or only allow for shallow interactions between them.

2.2 Dynamic Graph Neural Networks

In recent years, many Graph Neural Networks (GNNs), such as GCN (Kipf and Welling, 2016), GAT (Veličković et al., 2017), and GIN (Xu et al., 2019), have been developed to model complex relationships on graphs. These methods leverage the nodes and edges within the graph to model various real-world complex networks (Tian et al., 2022; He et al., 2024, 2023; Tang et al., 2023). However, models based on static graph often neglect the performance variations introduced by temporal evolution. Therefore, dynamic graph is more suitable for further exploring real-world applications. Among the various methods for modeling dynamic graphs, Discrete-Time Dynamic Graphs (DTDG) stand out as one of the most widely used approaches (Manessi et al., 2020; Zheng et al., 2023; Li et al., 2024). In the DTDG framework, the dynamic graph is represented as a sequence of graph snapshots, where each snapshot corresponds to the state of the graph at a particular discrete time step. At each time step, the graph can evolve in terms of its structure. Among these methods, a substantial amount of work focuses on snapshot updates and fusion in dynamic graphs. Pareja et al. 2020 updates the weight matrices of GCNs between snapshots. You et al. 2022 updates the node embeddings at different snapshot over time. Zhu et al. 2023 introduces a sliding window module to enhance the model's ability to capture dependencies over long sequences of snapshots. Different from the above works, we propose a FGDGNN that can capture fine-grained temporal information on the dynamic propagation graph, which fully utilizes the temporal information within the graph and the update information between graphs.

3 Methodology

3.1 Problem Definition

The rumor detection task can be defined as a classification problem. Formally, for a given rumor detection dataset $C = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_m\}$, where $\mathcal{C}_i$ is the i-th event and m is the number of events. For each event $\mathcal{C}_i = \{r_i, p^i_1, p^i_2, \ldots, p^i_{n_i-1}, G_i\}$, r_i is the source post, p^i_j represents the j-th responsive post, n_i is the number of posts in the event $\mathcal{C}_i$. All posts in event $\mathcal{C}_i$ are ordered chronologically and the set of timestamps for posts is denoted as $\mathcal{T}_i = \{t^i_0, t^i_1, t^i_2, \ldots, t^i_{n_i-1}\}$, where $t^i_0 = 0$ represents the timestamp of the source post and t^i_j represents the timestamp of the j-th responsive post. $G_i = \langle V_i, A_i, X_i \rangle$ is the propagation graph with the root node r_i, where V_i refers to the set of nodes corresponding to posts. $A_i \in \{0, 1\}^{n_i \times n_i}$ represents the adjacency matrix, where if there is a response relationship between node p^i_u and p^i_v, $A_{i(u,v)} = A_{i(v,u)} = 1$, otherwise $A_{i(u,v)} = A_{i(v,u)} = 0$. $X_i \in \mathbb{R}^{n_i \times d}$ denotes the node feature matrix , where d is the node embedding dimension. For simplicity, the subscript i is omitted in the following sections. Rumor detection aims to learn a function $f : C \to Y$ that classifies each event into one of the categories $Y \in \{F, T\}$ (i.e., Rumor or Non-Rumor).

3.2 Overview

In this section, we propose a novel Fine-Grained Dynamic Graph Neural Network (FGDGNN) model for rumor detection tasks. As illustrated in Figure 2, we will provide a detailed explanation for classifying rumors using FGDGNN, including Dynamic Graph Construction, Graph Representation and Graph-Level Fusion.

3.3 Dynamic Graph Construction

Formally, given a propagation event $\mathcal{C}$, we divide $\mathcal{C}$ into S graph snapshots along the timestamps of the source post and its responsive posts. Specifically, each graph snapshot is divided into equal time span $\Delta t = \frac{t_{n-1} - t_0}{S}$. After that, the propagation event $\mathcal{C}$ can be modeled as a dynamic propagation graph G which can be represented as a sequence of graph snapshots $G_s (s = 1, 2, \ldots, S)$. The later snapshot fully encompass the earlier snapshot, effectively simulating the dynamic evolution of propagation events. For each snapshot, we can obtain the propagation graph $G_s = \langle V_s, A_s, X_s \rangle$. $V_s = \{\mathcal{C}_s | \mathcal{T}_s \leq s\Delta t\}$ is the set of vertices, where

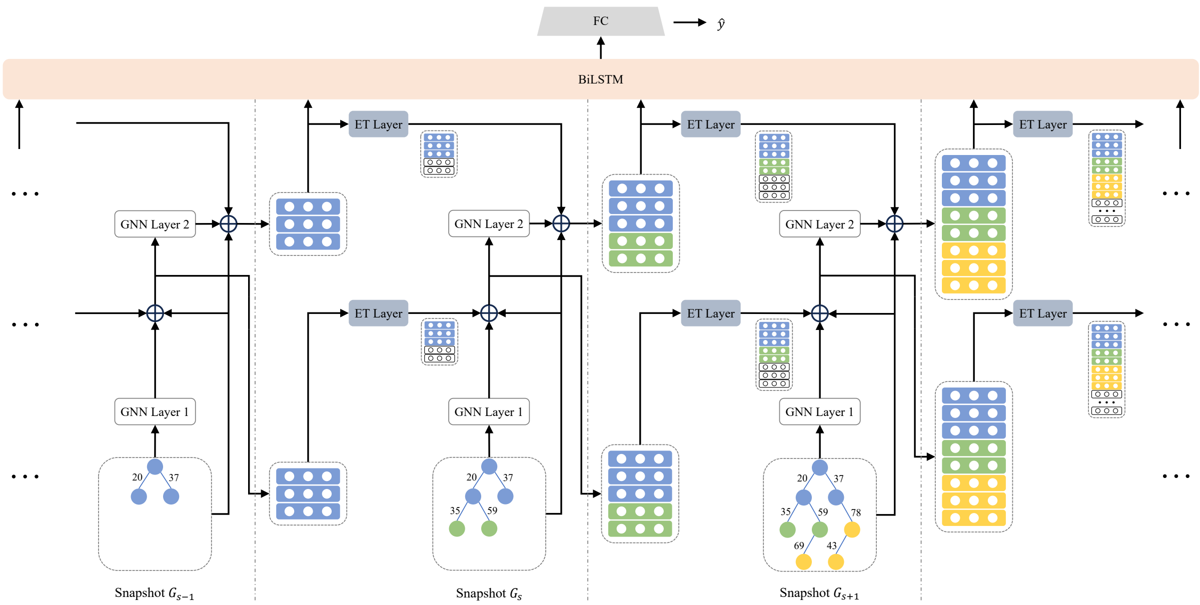


Figure 2: Overview of the proposed FGDGNN framework. The Embedding Transformation (ET) Layer represents the transformation of the node embedding dimensions.

each node has a timestamp. The time interval between when a post is created and when its responded post is created is used as edge weight. $A_s \in \{0,1\}^{n_s \times n_s}$ is the adjacency matrix. The node feature matrix denotes as $X_s \in \mathbb{R}^{n_s \times d}$, where d is the embedding vector dimension of each node.

3.4 Graph Representation

3.4.1 Temporal Information Encoding.

In the process of rumor spreading, the larger the time interval of a responsive post, the less attention it is likely to receive, along with a corresponding decrease in its influence and importance. We utilize the time intervals from posts to obtain temporal features and employ a decay mechanism $\varphi(t)$ to model the time interval in each snapshot.

$$\varphi(t) = \frac{1}{1 + \alpha \times (t - t_p)} \tag{1}$$

where t and t_p refer to the timestamps of the current post and its responded post respectively. α represents the decay factor.

Inspired by (Xu et al., 2020), we use the cosine function to encode decayed time information to capture the periodic variations in the time interval, and identify the propagation patterns of both rumors and non-rumors.

$$\omega(t) = cos(W_t \varphi(t) + b_t) \tag{2}$$

where W_t and b_t are learned parameters.

3.4.2 Edge-Aware Update.

We aim to learn the representations of the graph snapshots of dynamic propagation graph $G = \{G_1, G_2, \ldots, G_S\}$. As an effective graph structure, the Graph Isomorphism Network (GIN) (Xu et al., 2019) can capture the topological structure and node features of the graph and is suitable for rumor detection tasks. Given a graph snapshot G_s, the GIN encoder updates l-th layer the hidden feature vector $h_v^{(l)}$ for node v based on the $l-1$-th layer as follows:

$$h_v^{(l)} = \text{MLP}\left((1 + \epsilon^{(l)}) \cdot h_v^{(l-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(l-1)} \right) \tag{3}$$

GIN can effectively construct graph representations, but it does not incorporate temporal information (i.e., timestamps). Inspired by (Hu et al., 2019), to better leverage time interval, we adapt GIN to Edge-Aware GIN (EAGIN) defined in Equation (4) by introducing time interval to construct temporal dynamic graph. Specifically, We incorporate time interval as edge weights into the EAGIN to model the influence of each neighbor on the node.

$$\begin{aligned} h_v^{(l)} = \text{MLP}\Big(& (1 + \epsilon^{(l)}) \cdot h_v^{(l-1)} + \\ & \sum_{u \in \mathcal{N}(v)} \text{ReLU}(h_u^{(l-1)} * \omega(t)) \Big) \end{aligned} \tag{4}$$

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.

3.4.3 Node-Level Update.

The propagation graph G changes dynamically, and consequently, the node embeddings at different snapshots will also change. To capture dynamic node information in the evolving propagation graph, we propose a node-level embedding update mechanism. A two-layer EAGIN is employed, with node embeddings updated hierarchically in each hidden layer across snapshots.

$$H_s^{(1)} = \text{EAGIN}(A_s, H_s^{(0)}) \tag{5}$$

$$H_s^{(1)} = \beta\Phi_1(H_s^{(1)}) + (1-\beta)\Phi_1(H_{s-1}^{(1)'}) + \gamma\Phi_2(X_s) \tag{6}$$

$$H_s^{(2)} = \text{EAGIN}(A_s, H_s^{(1)}) \tag{7}$$

$$H_s^{(2)} = \beta\Phi_1(H_s^{(2)}) + (1-\beta)\Phi_1(H_{s-1}^{(2)'}) + \gamma\Phi_2(X_s) \tag{8}$$

where

$$H_{s-1}^{(1)'} = \text{ET Layer}(H_{s-1}^{(1)}) \tag{9}$$

$$H_{s-1}^{(2)'} = \text{ET Layer}(H_{s-1}^{(2)}) \tag{10}$$

and $H_s^{(0)} = X_s$. β and γ are learnable parameters. Φ_1 and Φ_2 represent Multi-Layer Perceptron (MLP). Through the ET Layer, the hidden state dimensions of the nodes generated by the previous snapshot are made consistent with the hidden state dimensions of the nodes in the current snapshot. Note that when the snapshot is the first one in the dynamic propagation graph, Equation (6) and Equation (8) do not include $H_{s-1}^{(l)}$.

We apply mean-pooling operators to obtain the representation g_s of the graph snapshot G_s. Finally, the dynamic propagation graph is represented as g.

$$g_s = \text{MEAN}(H_s^{(2)}) \tag{11}$$

$$g = \{g_1, g_2, \ldots, g_S\} \tag{12}$$

3.5 Graph-Level Fusion

After obtaining the graph representation of the dynamic propagation graph, we use Bidirectional Long Short-Term Memory (BiLSTM) (Hochreiter and Schmidhuber, 1997) to model the dependencies between snapshots. The forward and backward graph representation sequences are then used to capture the associations between snapshots. This process can be formally described as follows:

$$\begin{aligned}\overrightarrow{\mathbf{g}} &= \overrightarrow{\text{LSTM}}(g)\\ \overleftarrow{\mathbf{g}} &= \overleftarrow{\text{LSTM}}(g)\end{aligned} \tag{13}$$

Then we concatenate the forward state $\overrightarrow{\mathbf{g}}$ and the backward state $\overleftarrow{\mathbf{g}}$ to obtain the representations $\mathbf{g}$ encoded by BiLSTM, where CONCAT represents the concatenate operation.

$$\mathbf{g} = \text{CONCAT}(\overrightarrow{\mathbf{g}}, \overleftarrow{\mathbf{g}}) \tag{14}$$

3.6 Training Objective

To calculate the labels of the rumors, we apply a fully connected layer followed by a softmax layer,

$$\hat{y} = softmax(W_f\mathbf{g} + b_f) \tag{15}$$

where $\hat{y}$ is the predicted probability distribution. W_f and b_f are weight and bias parameters.

Our training object aims to minimize the cross-entropy loss $\mathcal{L}$ as follows:

$$\mathcal{L} = -\frac{1}{N}\sum_{b=1}^{N}\sum_{c=1}^{M} y_{b,c} log(\hat{y}_{b,c}) \tag{16}$$

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

Algorithm 1 illustrates the process of training propagation events using the proposed FGDGNN model.

Algorithm 1 Rumor detection algorithm

Input: the propagation event $\mathcal{C}$, the timestamps $\mathcal{T}$.
Output: the predicted probability distribution $\hat{y}$.

1: model the propagation event as a dynamic propagation graph G including a sequence of graph snapshots G_s;
2: **for** each snapshot G_s **do**
3:   obtain the temporal information $\omega(t)$ with Eq. 1 and Eq. 2;
4:   obtain the edge-ware and node-level representation H_s with Eq. 6 and Eq. 8;
5:   obtain the graph representation g_s with Eq. 11;
6: **end for**
7: obtain graph-level fusion $\mathbf{g}$ with Eq. 13;
8: producing predicted probability distribution $\hat{y}$ with Eq. 15;
9: update parameters in FGDGNN with Eq. 16;

Statistics	RumorEval	TWITTER	Weibo
# Events	245	1077	4310
# Posts	4145	60207	816217
# Non-Rumors	112	564	2187
# Rumors	133	513	2123
# Avg. time length	12 Hours	416 Hours	843 Hours

Table 1: Statistics of the datasets.

4 Experiments

4.1 Datasets

We evaluate the proposed model on three real-world rumor detection datasets: RumorEval (Derczynski et al., 2017), TWITTER (Lin et al., 2022) and Weibo (Ma et al., 2016). RumorEval and TWITTER are English datasets collected from the social media platform Twitter. Weibo is a Chinese dataset collected from the Sina Weibo. These three datasets are binary classification datasets, where each event is labeled as either Rumor (F) or Non-Rumor (T). In our experiments, the data used for each event includes the source post, responsive posts and the timestamp information of each post. We retain events with more than 3 comments in these datasets. Table 1 shows the statistics of the datasets.

4.2 Comparison Models

We compare the proposed model with the following baselines:

- **Bi-GCN** (Bian et al., 2020) is a rumor detection framework that models the top-down and bottom-up bi-directional GCN propagation.
- **EBGCN** (Wei et al., 2021) is an edge-enhanced rumor detection model that captures propagation structural features.
- **GACL** (Sun et al., 2022b) is a rumor detection model using adversarial and contrastive learning.
- **RDEA** (He et al., 2021) is a rumor detection framework that incorporates self-supervised learning and contrastive learning.
- **TrustRD** (Liu et al., 2023) is a rumor detection model that utilizes self-supervised pre-training and adversarial training.
- **DynGCN** (Choi et al., 2021) is a dynamic rumor detection framework that models graph snapshots and attention mechanisms.
- **PSGT** (Zhu et al., 2024) is a rumor detection framework that models graph transformer and propagation graph.

4.3 Experimental Setup

We follow the evaluation method in (Bian et al., 2020), and conduct 10 runs of 5-fold cross-validation to report the final results. The Accuracy (Acc.), Precision (Prec.), Recall (Rec.), and F1-score (F1) are adopted as evaluation metrics. The proposed model is implemented using PyTorch (Ketkar et al., 2021). Adam algorithm (Kingma and Ba, 2014) is used to optimize the parameters. The size of hidden layer is set to 128. The decay factor α is set to 1, 10 and 10 for RumorEval, TWITTER and Weibo respectively. The number of graph snapshots S in the dynamic graph generated for each event is set to 3.

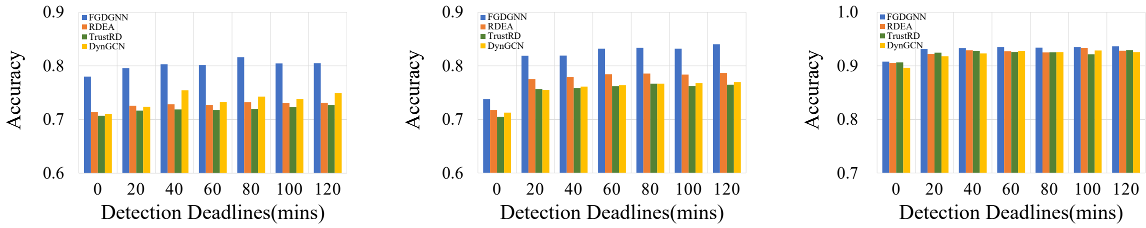
For RumorEval, TWITTER and Weibo datasets, following (Sun et al., 2022b; Ma et al., 2023), we join the source post with each comment post in a *[CLS] Source [SEP] Comment [SEP]* manner. BERT (Devlin et al., 2018) is employed to encode the posts and the final hidden state representation of *[CLS]* token is used as each corresponding node representation.

4.4 Results

Table 2 shows the results of rumor detection on three public real-world datasets. The experimental results demonstrate that the proposed FGDGNN model outperforms other baselines, which validates the effectiveness of modeling temporal information in the inter-snapshots and intra-snapshot. BiGCN only captures the spatial information of rumor events, making it vulnerable to rumor attacks. EBGCN uses edge weights to explore the potential relationships in a propagation graph. However, our proposed model, FGDGNN, employs time intervals as edge weights, enabling it to accurately capture the importance of nodes at different time points. Compared with standalone propagation structures, the performance of GACL, RDEA, and TrustRD in rumor detection tasks improves significantly when incorporating methods such as graph augmentation and contrastive learning. PSGT leverages the graph transformer to capture propagation structures and long-sequence dependencies. The models mentioned above focus on static graphs, whereas DynGCN models dynamic graphs. DynGCN uses various snapshot construction methods to investigate the task of rumor detection. However, the snap-

Method	Class	RumorEval				TWITTER				Weibo			
		Acc.	Prec.	Rec.	F1	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	0.9103	0.9099	0.9110	0.9104
	T		0.7042	0.6562	0.6794		0.7771	0.7732	0.7751		0.9147	0.9109	0.9128
EBGCN	F	0.6952	0.7100	0.7652	0.7366	0.7539	0.7449	0.7428	0.7438	0.9166	0.9151	0.9157	0.9154
	T		0.6812	0.6125	0.6450		0.7657	0.7640	0.7648		0.9176	0.9173	0.9175
GACL	F	0.7250	0.7974	0.7385	0.7668	0.7609	0.7987	0.6781	0.7335	0.9367	**0.9352**	0.9366	0.9359
	T		0.7591	0.7091	0.7332		0.7454	0.8377	0.7889		0.9386	**0.9369**	0.9378
RDEA	F	0.7321	0.7513	0.7890	0.7697	0.7855	0.7942	0.7496	0.7713	0.9340	0.9294	0.9378	0.9336
	T		0.7434	0.6648	0.7019		0.7857	0.8181	0.8016		0.9393	0.9303	0.9348
TrustRD	F	0.7267	0.7298	0.8175	0.7712	0.7695	0.7751	0.7359	0.7550	0.9312	0.9258	0.9355	0.9306
	T		0.7617	0.6195	0.6833		0.7749	0.7974	0.7860		0.9375	0.9266	0.9320
DynGCN	F	0.7377	0.7471	0.7617	0.7543	0.7693	0.7647	0.7495	0.7570	0.9274	0.9120	0.9244	0.9182
	T		0.7092	0.6823	0.6955		0.7759	0.7893	0.7825		0.9203	0.9075	0.9138
PSGT	F	0.8075	0.8209	**0.8361**	0.8285	0.8089	0.8148	0.7814	0.7977	0.9235	0.9171	0.9295	0.9233
	T		**0.8152**	0.7736	0.7939		0.8141	0.8332	0.8235		0.9315	0.9175	0.9244
FGDGNN	F	**0.8242**	**0.8653**	0.8054	**0.8343**	**0.8408**	**0.8454**	**0.8181**	**0.8315**	**0.9406**	0.9351	**0.9451**	**0.9401**
	T		0.7905	**0.8463**	**0.8175**		**0.8419**	**0.8617**	**0.8517**		**0.9466**	0.9361	**0.9413**

Table 2: Rumor detection results on three datasets. Abbrev.: Rumor (F), Non-Rumor (T).



(a) RumorEval dataset (b) TWITTER dataset (c) Weibo dataset

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

shots between DynGCN are isolated and lack any interconnections across snapshots. In contrast, our model emphasizes the fine-grained temporal information in the inter-snapshots by leveraging embedding updates across snapshots. In a snapshot, the time intervals are used as edge weights, and EAGIN is proposed to effectively capture the edge-aware temporal features in the intra-snapshot. The embedding update across snapshots integrates the node-level information in the inter-snapshots and can effectively capture the fine-grained temporal dependencies of propagation graph.

4.5 Ablation Study

In order to analyze the contribution of each module of our proposed model FGDGNN, we compare it with the variant models: (1) w/o Edge-Aware Update: removing the temporal information (i.e. time intervals) used as edge weights in each snapshot. (2) w/o Node-Level Update: removing the embedding update mechanism across snapshots. (3) w/o Edge-Aware Update & Node-Level Update: removing both time intervals and embedding update in dynamic graph. (4) w/o Dynamic: using static graph instead of dynamic graph. Specifically, we only use the last graph snapshots of the dynamic graph in the whole framework.

Table 3 presents the experimental results of the models on three datasets. Acc. refers to the overall results, while the F1-score refers specifically to the results for the Rumor (F) category. The experimental results show that removing any of the components leads to a decrease in the performance, demonstrating that each component plays an essential role in rumor detection. Specifically, when the time intervals are removed, the accuracy on RumorEval, TWITTER and Weibo drops by 1.21%, 0.74% and 0.59%. Time intervals record the temporal sequence of propagation event. When used as edge weights in graph neural networks, they help the model understand the importance of different responsive posts. Additionally, they capture the propagation patterns of rumors and non-rumors across different time periods, further aiding in the identification of the veracity of news event. If the embedding update mechanism is removed, the ac-

Model	RumorEval		TWITTER		Weibo	
	Acc.	F1	Acc.	F1	Acc.	F1
FGDGNN	**0.8242**	**0.8343**	**0.8408**	**0.8315**	**0.9406**	**0.9401**
w/o Edge-Aware Update	0.8121	0.8283	0.8334	0.8208	0.9347	0.9342
w/o Node-Level Update	0.8163	0.8311	0.8309	0.8216	0.9311	0.9307
w/o Edge-Aware Update & Node-Level Update	0.8042	0.8237	0.8218	0.8089	0.9303	0.9295
w/o Dynamic	0.7804	0.8125	0.8129	0.7851	0.9332	0.9322

Table 3: Results of ablation study on three datasets.

curacy on three datasets drops by 0.79%, 0.99% and 0.95%. As time progresses, the propagation states of different snapshots evolve. By employing a embedding update mechanism across snapshots, the model can capture the dynamic evolution patterns of the propagation process, enabling information transfer across time steps and enhancing the model's memory of historical data. When the time intervals and embedding update are removed at the same time, the accuracy on three datasets drops by 2%, 1.9% and 1.03%. The time intervals within each snapshot, along with the dynamic embedding update mechanism across snapshots, effectively captures the fine-grained temporal dependencies between nodes. When dynamic graphs are replaced with static graphs, the accuracy on three datasets drops by 4.38%, 2.79% and 0.74%. Modeling the propagation structure as a dynamic graph enables more accurate capture of the temporal features and dynamic evolution of information spread, thereby improving detection accuracy. This approach offers a clear advantage over static graphs.

4.6 Different GNNs Components

Table 4 shows the experimental results of using different graph neural networks as graph encoders. It is observed that EAGIN in our proposed FGDGNN model yields the best performance. EAGIN uses the temporal information in the inter-snapshots, which can better capture the graph structure and distinguish the spread of rumors and non-rumors than GIN. In contrast, GAT focuses on neighboring nodes via a self-attention mechanism that assigns weights based on local neighborhood information. The ability of processing global graph structure is limited compared with GIN. GCN tends to aggregate information from neighboring nodes in a way that leads to excessive smoothing, which can reduce its expressive power and hinder its ability to capture deeper or more complex propagation patterns. The experimental results demonstrate that EAGIN is better that others in improving the effectiveness of the FGDGNN model.

Model	RumorEval	TWITTER	Weibo
	Acc.	Acc.	Acc.
EAGIN	**0.8242**	**0.8408**	**0.9406**
GIN	0.8121	0.8334	0.9347
GAT	0.8017	0.8266	0.9339
GCN	0.7938	0.8215	0.9333

Table 4: Results of different GNN on three datasets.

4.7 Early Rumor Detection

This experiment aims to detect rumors on social media at an early stage, facilitating early rumor detection. To construct the detection task, we follow the methodology in (Sun et al., 2022b), setting a series of detection deadlines. Figure 3 illustrates the performance of FGDGNN in early rumor detection, comparing it with RDEA, TrustRD and DynGCN across various deadlines on three datasets. It can be observed that at time 0, all models perform poorly due to the limited training data caused by a lack of responsive post. Subsequently, as the detection deadline rises, the accuracy of all models is improved. Notably, FGDGNN consistently achieves higher accuracy than the other models at every deadline, demonstrating its superior performance in early rumor detection.

5 Conclusion

In this paper, we propose a novel Fine-Grained Dynamic Graph Neural Network (FGDGNN) model for rumor detection. We construct the edge-weighed propagation graph in which the time intervals are used as edge weights each snapshot. Additionally, we propose an embedding transformation layer to update node embeddings across snapshots. Experiments on three public datasets demonstrate that FGDGNN model outperforms the state-of-the-art baselines.

Limitations

One limitation of our model is that the constructed temporal information does not account for multi-scale temporal encoding. If the dynamic changes of an event are associated with different time scales (such as minutes, hours, or days), it may lead to suboptimal performance. In the future, we will explore more approaches for temporal modeling to enhance the performance of rumor detection further.

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.

Ya-Ting Chang, Zhibo Hu, Xiaoyu Li, Shuiqiao Yang, Jiaojiao Jiang, and Nan Sun. 2024. Dihan: A novel dynamic hierarchical graph attention network for fake news detection. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 197–206.

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*.

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.

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.

Bowei He, Xu He, Yingxue Zhang, Ruiming Tang, and Chen Ma. 2023. Dynamically expandable graph convolution for streaming recommendation. In *Proceedings of the ACM Web Conference 2023*, pages 1457–1467.

Buyun He, Yingguang Yang, Qi Wu, Hao Liu, Renyu Yang, Hao Peng, Xiang Wang, Yong Liao, and Pengyuan Zhou. 2024. Dynamicity-aware social bot detection with dynamic graph transformers. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*.

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.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. 2019. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265*.

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*.

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

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.

An Lao, Chongyang Shi, and Yayi Yang. 2021. Rumor detection with field of linear and non-linear propagation. In *Proceedings of the Web Conference 2021*, pages 3178–3187.

Hongxi Li, Zuxuan Zhang, Dengzhe Liang, and Yuncheng Jiang. 2024. K-truss based temporal graph convolutional network for dynamic graphs. In *Asian Conference on Machine Learning*, pages 739–754. PMLR.

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.

Hongzhan Lin, Jing Ma, Liangliang Chen, Zhiwei Yang, Mingfei Cheng, and Chen Guang. 2022. Detect rumors in microblog posts for low-resource domains via adversarial contrastive learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2543–2556.

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.

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. 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.

Franco Manessi, Alessandro Rozza, and Mario Manzo. 2020. Dynamic graph convolutional networks. *Pattern Recognition*, 97:107000.

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.

Aldo Pareja, Giacomo Domeniconi, Jie Chen, Tengfei Ma, Toyotaro Suzumura, Hiroki Kanezashi, Tim Kaler, Tao Schardl, and Charles Leiserson. 2020. Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5363–5370.

Chenguang Song, Kai Shu, and Bin Wu. 2021. Temporally evolving graph neural network for fake news detection. *Information Processing & Management*, 58(6):102712.

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.

Haoran Tang, Shiqing Wu, Guandong Xu, and Qing Li. 2023. Dynamic graph evolution learning for recommendation. In *Proceedings of the 46th international acm sigir conference on research and development in information retrieval*, pages 1589–1598.

Xiang Tao, Liang Wang, Qiang Liu, Shu Wu, and Liang Wang. 2024. Semantic evolvement enhanced graph autoencoder for rumor detection. In *Proceedings of the ACM on Web Conference 2024*, pages 4150–4159.

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.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

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.

Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. 2020. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*.

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.

Shuai Xu, Jianqiu Xu, Shuo Yu, and Bohan Li. 2024. Identifying disinformation from online social media via dynamic modeling across propagation stages. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2712–2721.

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.

Jiaxuan You, Tianyu Du, and Jure Leskovec. 2022. Roland: graph learning framework for dynamic graphs. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 2358–2366.

Shangfei Zheng, Hongzhi Yin, Tong Chen, Quoc Viet Hung Nguyen, Wei Chen, and Lei Zhao. 2023. Dream: Adaptive reinforcement learning based on attention mechanism for temporal knowledge graph reasoning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1578–1588.

Junyou Zhu, Chao Gao, Ze Yin, Xianghua Li, and Jürgen Kurths. 2024. Propagation structure-aware graph transformer for robust and interpretable fake news detection. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4652–4663.

Yifan Zhu, Fangpeng Cong, Dan Zhang, Wenwen Gong, Qika Lin, Wenzheng Feng, Yuxiao Dong, and Jie Tang. 2023. Wingnn: Dynamic graph neural networks with random gradient aggregation window. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3650–3662.