DNKGAT: Dynamic Neighbor-enhanced Knowledge Graph Attention Network for Rumor Detection

Anonymous ACL submission

Abstract

Detecting rumors on social media is critical due to their rapid spread and harmful effects, yet existing models often overlook integrating spatial and temporal neighboring information of message propagation, as well as the dynamics of background knowledge in user comments. To address this gap, we present a principled *Dynamic Neighbor-enhanced Knowledge Graph Attention Network* (DNKGAT), which unifies the dynamics of message propagation and evolving background knowledge from knowledge graphs. Specifically, the proposed method employs a multi-hop knowledge graph attention mechanism to incorporate extensive neighboring information from knowledge graphs, a feature previously underexplored. The framework includes a post-enhancement unit and a rumor classification module, enhancing detection capabilities by learning dynamic event representations and aggregating them progressively to capture cascading effects for more effective rumor identification. Extensive experiments on two real-world datasets demonstrate significant improvements over strong baselines, particularly in early-stage rumor detection. Our implementation available at https://anonymous.4open.science/r/DNKGAT-FC6C.

1 Introduction

The Internet and social media platforms like Twitter and Facebook have become essential ways for people to access news in their daily lives. These platforms enable the rapid and free dissemination of news, allowing the public to express opinions and communicate freely. However, the lack of effective censorship results in lower-quality news compared to traditional methods, leading to a noisy information ecosystem plagued with disinformation and rumors. Therefore, it is of prominent importance to detect rumors on social media. Currently, three prominent methods are utilized for rumor detection:

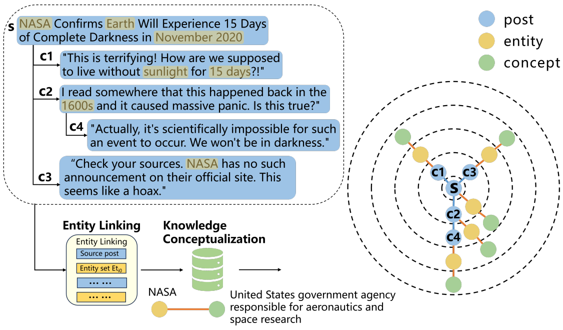


Figure 1: The introduction of the proposed method.

those based on spatial structure, temporal structure, and knowledge graph.

Rumor detection aims to identify rumors automatically. Recent studies focus on the dissemination process of news articles, where users engage by posting, reposting, or responding to specific articles. These interactions create tree-based (Ma et al., 2018a; Ma and Gao, 2020; Bian et al., 2020) or graph-based (Huang et al., 2019; He et al., 2021; Sun et al., 2022b) structures. By analyzing the structure of news propagation and assessing user trustworthiness, it is possible to deduce the likely veracity of the news. Some studies emphasize the importance of temporal structure, enabling the modeling of fine-grained dynamic features and enhancing early detection performance (Ma et al., 2015; Choi et al., 2021; Song et al., 2021b). Considering both spatial and temporal structures in message propagation is crucial. Several studies incorporate external knowledge to improve rumor detection (Zhang et al., 2019; Wang et al., 2020; Hu et al., 2021), with their effectiveness analyzed in (Hinkelmann et al., 2022). However, such extensive spatial and temporal neighboring information of message propagation from knowledge graph is not exploited by existing models.

This paper aims to model the spatial-temporal structure of messages and associated background knowledge within a unified framework for timely rumor detection. Traditional methods, relying on graph neural networks (GNNs) and graph convolutional networks (GCNs) with a message-passing framework, learn spatial features of rumors but struggle to aggregate high-order neighboring node information without stacking multiple layers when dealing with deeper node relationships. To this end, we introduce the Dynamic Neighbor-enhanced Knowledge Graph Attention Network (DNKGAT), featuring a dynamic multi-hop knowledge graph attention mechanism that captures spatial, temporal, and knowledge information through evolving messages and knowledge graphs. In Figure 1, source post and comments (blue nodes) are linked to entities (yellow nodes) and concepts (green nodes) via entity linking and knowledge conceptualization. One entity (**c2**) can be involved in multiple triplets, serving as the contributor enriching entities' feature and propagation information. This mechanism enables source posts to aggregate high-order neighbor information, like **c4**'s entity and concept. We also propose a fusing method to enhance these representations with the message, using post-enhancement to concatenate dynamic neighbor-enhanced knowledge graph attention network representations with initial message representations, allowing for incremental learning of better event representations prior to rumor classification. The main contributions are summarized as follows:

- We propose DNKGAT, a pioneering approach that captures spatial-temporal rumor characteristics through an incremental learning process and a post-enhancement unit for improved detection accuracy.
- Leveraging a multi-hop knowledge graph attention mechanism, DNKGAT aggregates extensive neighboring information to capture message propagation dynamics and integrate background knowledge for comprehensive contextual understanding in rumor detection.
- DNKGAT introduces the first post-enhancement unit, which learns feature interactions between posts and knowledge through a cross-information sharing layer to enrich information and improve detection.
- Experiment results show that the proposed method outperforms the strong baselines on two real-world datasets and can effectively detect rumors at an early stage.

2 Related Work

2.1 Spatial Structure Based Rumor Detection

Spatial structure-based methods in rumor detection model news propagation paths to differentiate real and fake news dissemination patterns on social networks. Ma et al. (2018a) uses a tree-based recursive neural network for content semantics and propagation cues, while Huang et al. (2019) employs GCNs for spatial structure capture. Bian et al. (2020) enhances rumor dissemination understanding with Bi-GCNs, and Guo et al. (2023) integrates GNNs with convolutional and recurrent neural networks for semantic capture. Liu et al. (2024) combines GCN and attention mechanisms for influence and propagation structure relations. Despite effectiveness, these methods often overlook temporal dynamics of message propagation, missing critical patterns for rumor identification.

2.2 Temporal Structure Based Rumor Detection

Temporal structure-based methods in rumor detection model the temporal information of news propagation to differentiate real and fake news dissemination patterns. Ma et al. (2016) uses a recurrent neural network-based model for semantic variations, while Liu and fang Brook Wu (2018) combines recurrent and convolutional networks for temporal structure. Song et al. (2021a) models real-world news evolution under continuous-time dynamic diffusion networks, and Sun et al. (2022a) unifies message propagation dynamics with background knowledge from knowledge graphs. Gong et al. (2023) integrates a neural Hawkes process for self-exciting patterns of true and fake news. Despite their effectiveness, most methods overlook integrating external knowledge, which could significantly improve rumor detection.

2.3 Knowledge Based Rumor Detection

Knowledge-based methods in rumor detection exploit external knowledge to supplement post content for identification. Zhang et al. (2019) complements short texts with external knowledge for improved rumor detection, while Cui et al. (2020) uses an article-entity bipartite graph and a medical knowledge graph for better news embeddings. KMGCN (Wang et al., 2020) models global structures among texts, images, and knowledge concepts for comprehensive semantic representations. KAN (Dun et al., 2021) validates knowledge attention effectiveness for rumor detection. Despite their benefits, these methods often overlook the contradiction between rumor content and external knowledge, and the importance of high-order neighboring knowledge, which this study addresses. CompareNet (Hu et al., 2021) compares news articles to a knowledge base through entities for detection. However, these methods overlook the importance of high-order neighboring knowledge, which this study fully leverages to improve its effectiveness.

3 Methodology

3.1 Problem Definition

The task of rumor detection can be defined as a binary classification problem aimed at detecting rumor items, e.g. Twitter posts. Let $\mathcal{O} = \{o_1, \ldots, o_n\}$ be a set of events, where each event o_i consists of a sequence of posts and comments. We denote the set of post and comment contents for each event as $o_i^c = \{c_{i0}, c_{i1}, \ldots, c_{im_i-1}\}$, where c_{i0} is the source post s_i, and c_{ij} is the j-th comment. m_i represents the number of posts and comments in o_i. Additionally, we define the relative release time sequence $o_i^t = \{t_{i0}, t_{i1}, \ldots, t_{im_i-1}\}$, where $t_{i0} = 0$ and t_{ij} denotes the release time of the j-th comment. The combined representation of each event is $o_i = \{(c_{i0}, t_{i0}), (c_{i1}, t_{i1}), \ldots, (c_{im_i-1}, t_{im_i-1})\}$.

Each event o_i is segmented into γ stages based on its duration, where γ is a hyperparameter. Each stage r has an equal time interval $\Delta t_i = \frac{t_{im_i-1}}{\gamma}$. The r-th sub-event of o_i is defined as $o_{ir} = \{(c_{i\pi}, t_{i\pi}) \mid t_{i\pi} \leq r\Delta t_i\}$.

The goal is to learn a model $f : \mathcal{O} \rightarrow \mathcal{Y}$ to classify each event o_i into predefined categories $\mathcal{Y} = \{0, 1\}$, where 0 denotes non-rumor and 1 denotes rumor.

For ease of understanding, the important mathematical notations used throughout the paper are listed in Table 1.

Notations	Descriptions
o_i	the i-th event
o_i^c	post sequence of the i-th event
o_i^t	relase time sequence of the i-th event
γ	the number of time stages
Δt_i	equal time interval of event o_i
o_{ir}	the r-th stage subset of o_i
Et_i	the entity set of o_i
Ct_i	the concept set of o_i
$G_{ir}^k = \langle V_{ir}^k, E_{ir}^k \rangle$	knowledge graph at stage r

Table 1: Important notations and descriptions

3.2 Overview

The overall architecture of our rumor detection approach is presented in Figure 2. It consists of four modules: (1) **dynamic knowledge graph construction:** constructs a dynamic post-entity-concept tripartite knowledge graph by integrating post-related external knowledge; (2) **dynamic neighbor-enhanced knowledge graph attention network**: generates node representations that integrate spatial, temporal, and knowledge information through a multi-hop attention mechanism, capturing the evolving relationships and context essential for accurate rumor detection; (3) **post-enhancement unit**: enriches the knowledge information by learning the feature interactions between the post and knowledge; (4) **rumor classifier**: aggregates the final knowledge and source post textual information to determine whether the event is a rumor or not.

3.3 Dynamic Knowledge Graph Construction Module

This module constructs a dynamic post-entity-concept tripartite knowledge graph to capture the evolving relationships between posts, entities, and concepts, thereby enriching the semantic understanding necessary for accurate rumor detection.

Posts often contain condensed content with numerous entity mentions, which can be ambiguous due to aliases, abbreviations, and alternative spellings. For example, in the post "*Exciting updates from @Tesla about their new Model S enhancements. Can't wait for the test drive!*", it is crucial to discern that "*Tesla*" refers to an "*electric vehicle manufacturer*" and "*Model S*" is a "*specific model of electric car*". To address this, we integrate external knowledge from both source posts and their comments, which evolves as the event progresses.

Entity linking and conceptualization extract rich semantic information from the posts. Using the NLTK for entity linking, we connect entity mentions to corresponding entities in a knowledge graph like YAGO. We drive conceptual information from YAGO based on the "isA" relationship, which defines the connection between entities and their concepts. For instance, for the entities identified in the example post, we obtain $Concept_{Tesla}$ = {Vehicle Manufacturer, Technology Company} $Concept_{Model\ S}$ = {Electric Car, Automobile}.

For each time stage o_r, we construct a dynamic

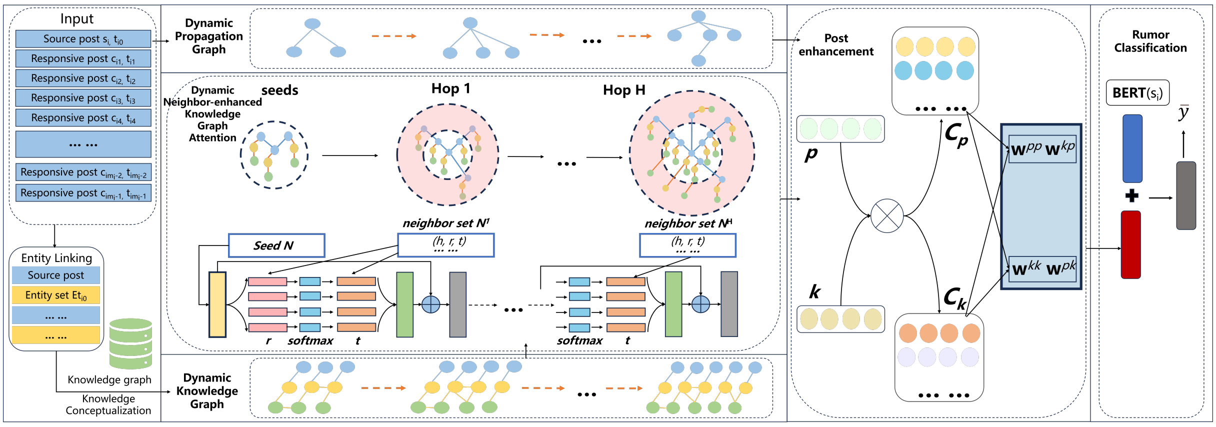


Figure 2: The framework of the proposed method. It consists of four components: (1) the leftest block: the input of the model; (2) the lefter block: dynamic neighbor-enhanced knowledge graph attention network; (3)the righter block: the post-enhancement module to enrich representations with initial feature; (4) the rightest block: a rumor classification module after post enhancement.

knowledge graph $\{G_{i1}^k, \ldots, G_{i\gamma}^k\}$, where $G_{ir}^k = \langle V_{ir}^k, E_{ir}^k \rangle$ includes vertices from posts, comments, entities, and concepts. Edges are established based on:

- **Post-entity edges:** These are formed between a post and an entity if the post contains a mention of the entity. The edge weight is determined by the term frequency-inverse document frequency(TF-IDF) of the entity within the post.
- **Entity and concept edges:** The relationships between entities, and between entities and concepts, are quantified using Pointwise Mutual Information(PMI), calculated over a fixed-size sliding window from a global corpus.

The adjacency matrix A_{ir}^k of the dynamic knowledge graph G_{ir}^k retains only those edges with significant TF-IDF or positive PMI scores. Each node v in the graph is initialized with its word embedding vector $k \in \mathbb{R}^F$, facilitating the propagation of semantic information through the graph.

3.4 Dynamic Neighbor-enhanced Knowledge Graph Attention Network

This module generates node representations incorporating spatial, temporal, and knowledge information. By exploiting the idea of graph attention network (Velickovic et al., 2017), we generate attentive weights of cascaded propagations to reveal the importance of connectivity. Here we start by describing a single layer, which consists of information propagation and aggregation, and then discuss how to generalize it to multiple layers.

Information Propagation: One entity can be involved in multiple triplets, serving as the contributor enriching entities' feature and propagation information. Taking $p_1 - e_1 - c_1$ and $p_1 - e_1 - c_2$ as an example, entity e_1 can take attributes c_1 and c_2 to enrich its features and then contribute post p_1's features, which can be simulated by propagating information from c_1 and c_2 to p_1. To characterize nodes' hierarchically extended propagation in terms of KG, in our methods, we recursively define the set of k-hop relevant entities for r-th time stage knowledge graph as follows:

Definition 1 (Relevant Entity) *Given the knowledge graph $\mathcal{G}_i$ in i-th time stage, the set of k-hop relevant entities for graph $\mathcal{G}_i$ is defined as:*

$$\mathcal{E}_i^k = \{t \mid (h, r, t) \in \mathcal{G}_i \quad and \quad h \in \mathcal{E}_i^{k-1}\} \\ k = 1, 2, 3, ..., H. \tag{1}$$

where $\mathcal{E}_i^0 = V_i$ is the set of vertex at the beginning of i-th stage, which can be seen as the seed set of the knowledge graph.

Relevant entities can be regarded as natural extensions of the knowledge graph. Given the definition of relevant entities, we then define the k-hop neighbor set of knowledge graph as follows:

Definition 2 (Neighbor Set) *The k-hop neighbor set of knowledge graph $\mathcal{G}_i$ is defined as the set of knowledge triples starting from $\mathcal{E}_i^{k-1}$:*

$$\mathcal{N}_i^k = \{(h, r, t) \mid (h, r, t) \in \mathcal{G}_i \quad and \quad h \in \mathcal{E}_i^{k-1}\} \\ k = 1, 2, 3, ..., H. \tag{2}$$

One concern about neighbor sets is their sizes may get too large with the increase of hop number k. The number of maximal hop H is usually not too large in practice, since nodes that are too distant from i-th KG may bring more noise than positive signals. We will discuss the size of neighbor set $\mathcal{S}$ in the experiments part. In our method, we can sample a fixed-size set of neighbors instead of using a full neighbor set to further reduce the computation overhead.

Given the i-th time stage node embedding h_i and the 1-hop neighbor set $\mathcal{N}_i^1$, each triple (h_i, r_i, t_i) is assigned a relevance probability by comparing i-th KG node embedding k_i to the i-th stage node h_i in $\mathcal{N}_i^1$ and the corresponding relation r_i in this triple:

$$p_i = \text{softmax}({k_i}^T r_i h_i) = \frac{\exp({k_i}^T r_i h_i)}{\sum_{(h,r,t)\in\mathcal{N}_i} \exp({k_i}^T r h)} \tag{3}$$

where $r_i \in \mathbb{R}^d$ and $h_i \in \mathbb{R}^d$ are the embeddings of the relation r_i and head h_i, respectively. The relevance probability p_i controls the decay factor on each propagation on edge (h_i, r_i, t_i), indicating how much information being propagated from t_i to h_i conditioned to relation r_i, regarded as the similarity of node k_i and the entity h_i measured in the space of relation r_i. Furthermore, when performing propagation forward, the p_i suggests parts of the data to focus on, which can be treated as explanations behind the propagation.

After obtaining the relevance probabilities, we take the sum of tails in $\mathcal{N}_i^1$ weighted by the corresponding relevance probabilities, and the vector $k_{\mathcal{N}_i^1}$ is returned:

$$k_{\mathcal{N}_i^1} = \sum_{(h_i,r_i,t_i)\in\mathcal{N}_i^1} p_i t_i, \tag{4}$$

where $t_i \in \mathbb{R}^d$ is the embedding of tail t_i.

Distinct from the information propagation in GCN (Kipf and Welling, 2016) and GraphSage (Hamilton et al., 2017) which set the discount factor between two nodes as a fixed number, our model not only exploits the proximity structure graph but also specifies varying importance of neighbors. Moreover, distinct from the graph attention network which only takes node representations as inputs, we model the relation r between h and t, encoding more information during propagation.

Information Aggregation: This phase is to aggregate the node representation k_i and its 1-hop neighbor set representations $k_{\mathcal{N}_i^1}$ - more formally, $k_i^{(1)} = f(k_i, k_{\mathcal{N}_i^1})$. We implement $f(\cdot)$ using the Bi-Interaction Aggregator (Wang et al., 2019b) to consider two kinds of feature interactions between k_i and $k_{\mathcal{N}_i^1}$, as follows:

$$\begin{aligned} f_{\text{Bi-Interaction}} = &\text{LeakyReLU}(W_1(k_i + k_{N_i^1})) + \\ &\text{LeakyReLU}(W_2(k_i \odot k_{N_i^1})), \end{aligned} \tag{5}$$

where $W_1, W_2 \in \mathbb{R}^{d' \times d}$ are trainable weight matrices, and $\odot$ denotes the element-wise product. Distinct from GCN and GraphSage aggregators, we additionally encode the feature interaction between k_i and $k_{\mathcal{N}_i^1}$. This term makes the information being propagated sensitive to the affinity between k_i and $k_{\mathcal{N}_i^1}$, e.g., passing more messages from similar entities.

We further stack more propagation layers to explore the high-order connectivity information, gathering the information propagated from the high-hop neighbors. More formally, in the l-th steps, we recursively formulate the representation of KG as:

$$k_i^{(l)} = f(k_i^{(l-1)}, k_{\mathcal{N}_i^{(l-1)}}) \tag{6}$$

3.5 Post-Enhancement Unit

Enlightened by the idea of root feature enhancement in (Bian et al., 2020), the source post of an event is pivotal in rumor detection. Inspired by the (Wang et al., 2019a) and (Guo et al., 2020) methods, we have designed the cross-information sharing layer, aiming to combine the strengths of both approaches, which can concatenate the latent feature vector of each node learned from the last dynamic neighbor-enhanced knowledge graph attention network with the initial feature vector of each post p in o_i^c learned from the word embedding.

For a knowledge node k and its corresponding post node p, we first construct $d \times d$ pairwise interactions of their latent feature $k \in \mathbb{R}^d$ and $p \in \mathbb{R}^d$:

$$C_k = kp^T = \begin{bmatrix} k^{(1)}p^{(1)} & \cdots & k^{(1)}p^{(d)} \\ \vdots & \ddots & \vdots \\ k^{(d)}p^{(1)} & \cdots & k^{(d)}p^{(d)} \end{bmatrix}, \tag{7}$$

$$C_p = pk^T = \begin{bmatrix} p^{(1)}k^{(1)} & \cdots & p^{(1)}k^{(d)} \\ \vdots & \ddots & \vdots \\ p^{(d)}k^{(1)} & \cdots & p^{(d)}k^{(d)} \end{bmatrix}, \tag{8}$$

where $C_k \in \mathbb{R}^{d\times d}$ represents the cross feature matrix from knowledge to post, $C_p \in \mathbb{R}^{d\times d}$ represents the cross feature matrix from post to knowledge and d is the dimension of hidden layers.

We then output the feature vectors of knowledge and post for the classification by projecting the cross feature matrix into their latent representation spaces:

$$\begin{aligned} k &= C_k w_1 + b_1 \\ &= kp^T w_1 + b_1, \end{aligned} \tag{9}$$

$$\begin{aligned} p &= C_p w_2 + b_2 \\ &= pk^T w_2 + b_2, \end{aligned} \tag{10}$$

where w and b are trainable weight and bias vectors. The weight vectors project the cross feature matrix from $\mathbb{R}^{d\times d}$ space back to the feature spaces $\mathbb{R}^d$. Through post-enhancement units, our method can enhance the post feature after dynamic knowledge graph operation.

3.6 Rumor Classification Module

In this module, we introduce the rumor classification module. It deploys a series of fully connected layers followed by sigmoid activation to predict whether the posts are fake or real. The module is built on top of the post-enhancement unit, thus taking the node feature representation as input. We leverage mean pooling to distill node representations $H_{i\gamma}$ from the post-enhancement module into a single vector H_i for each post:

$$H_i = \mathrm{MEAN}(H_{i\gamma}) \tag{11}$$

Then, we merge this with the BERT-extracted features (Devlin et al., 2019) of the source post s_i into a unified representation $\tilde{S}$:

$$\tilde{S} = \mathrm{LeakyReLU}(linear(H_i || BERT(s_i))) \tag{12}$$

A series of fully connected layers, followed by a sigmoid activation, is applied to $\tilde{S}$ to predict the rumor label $\hat{y}_i$:

$$\hat{y}_i = \sigma(w_f \tilde{S} + b_f) \tag{13}$$

where w_f and b_f are the weight and bias parameters. We then use cross entropy loss as the rumor classification loss:

$$\mathcal{L}_c = -\sum_i y_i \log \hat{y}_i \tag{14}$$

where y_i is the ground truth label of the i-th instance.

4 Experiments

4.1 Datasets

We experiment with two public Twitter datasets, Pheme5 and Pheme9, where each event is labeled as rumor or non-rumor. Pheme5 includes rumor tweets from five major events: Charliehebdo, Ferguson, Germanwings-crash, Otawashooting, and Sydney-siege, each with numerous sub-events containing source posts, responsive posts, propagation structures, and posting times. Pheme9 adds four events maintaining the same structure. Dataset statistics are detailed in Table 2.

Statistics	Pheme5	Pheme9
# of Posts	103,212	105,354
# of events	5,802	6,425
# of Non-rumors	3,830	4,023
# of Rumor	1,972	2,402
# of classes	2	2
Avg. # of words/ post	13.6	13.6
Avg. # of posts/ event	17.8	16.3
Max # of posts/ event	346	246
Min # of posts/ event	1	1

Table 2: Statistics of Datasets

4.2 Comparison Methods

We compare with the following baselines:

SVM-BOW (Ma et al., 2018b): employs a bag-of-words model for feature representation and utilizes a Support Vector Machine (SVM) as the classification algorithm.

CNN (Chen et al., 2017): employs a convolutional neural network for feature extraction from posts and applies a softmax function as the classification layer.

BiLSTM (Augenstein et al., 2016): leverages a bidirectional long short-term memory (Bi-LSTM) network to capture the contextual information within posts.

BERT (Devlin et al., 2019): is a pre-trained language model that utilizes bidirectional transformers. We use it to obtain the representation of the source post for classification.

CSI (Ruchansky et al., 2017): is composed of three modules: Capture, Score, and Integrate, and incorporates the behavior of both parties, users and articles, and the group behavior of users who propagate fake news.

DEFEND (Shu et al., 2019): uses a sentence-comment co-attention sub-network to exploit both news contents and user comments to jointly capture

Method	Pheme5				Pheme9			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
SVM-BOW	0.669	0.535	0.524	0.529	0.688	0.518	0.512	0.515
CNN	0.787	0.737	0.702	0.719	0.795	0.731	0.673	0.701
BiLSTM	0.795	0.763	0.691	0.725	0.794	0.727	0.677	0.701
BERT	0.865	0.859	0.851	0.855	0.844	0.834	0.835	0.835
CSI	0.857	0.843	0.859	0.851	0.851	0.836	0.855	0.845
DEFEND	0.868	0.867	0.859	0.863	0.863	0.857	0.859	0.858
RDM	0.873	0.817	0.823	0.820	0.858	0.847	0.859	0.852
DDGCN	0.846	0.844	0.817	0.823	0.855	0.846	0.841	0.843
DNKGAT	**0.892**	**0.882**	**0.877**	**0.879**	**0.894**	**0.889**	**0.887**	**0.888**

Table 3: Results of comparison among different models on Pheme5 and Pheme9 Datasets. We run the models five times, and report average results here.

explainable top-k check-worthy sentences and user comments for fake news detection.

RDM (Zhou et al., 2019): uses reinforcement learning to detect rumors early, determining the minimum posts needed for classification.

DDGCN (Sun et al., 2022a): is a dual dynamic graph convolutional network. It can learn the dynamics of messages in propagation and the dynamics of background knowledge from Knowledge graphs simultaneously.

4.3 Experiment Setup

We adopt the default optimization settings reported in corresponding papers for all comparison methods. We implement our method with Pytorch framework (Paszke et al., 2019). We set the number of time stages $\gamma = 3$. The number of epochs is 5. The parameters are optimized using the Adam algorithm. BERT-base (Devlin et al., 2019) is used as the encoder for the source post and pre-trained on the datasets. We split the Pheme5 dataset and Pheme9 dataset into training, validation, and testing sets with a split ratio of 7:1:2 without overlapping. We select the best parameter settings based on the performance of the validation set. We employ Accuracy, Precision, Recall, and F1 as evaluation metrics. We randomly split the datasets into five parts, and conduct 5-fold cross-validation to obtain the final results.

4.4 Performance Analysis

Table 3 shows the performance of the compared models. On both datasets, our model significantly outperforms all the other approaches in all the metrics, which confirms that considering the dynamic neighboring information would benefit the rumor detection task.

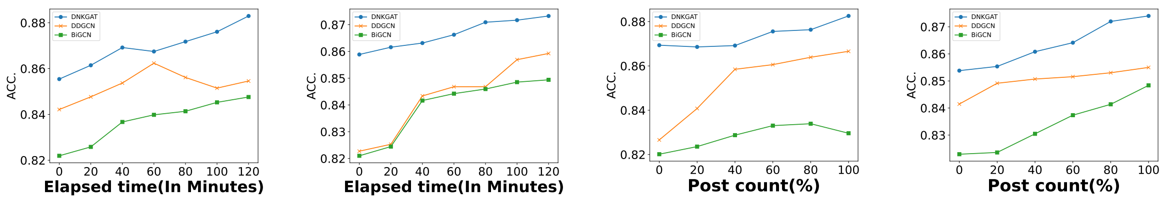
SVM-BOW underperforms due to its reliance on coarse, handcrafted features, lacking the nuanced capture required for broad generalization. In contrast, deep learning models, such as CNN, BiLSTM, and BERT, excel in automatically extracting effective features, with BERT particularly demonstrating superior semantic capture for rumor detection. DDGCN further outperforms most text and spatial models, suggesting that knowledge features offer complementary benefits. Our approach, compared to baselines, integrates temporal and spatial structures with external knowledge from source posts and comments, adaptively adjusting node influence in propagation structures to focus on key information, and emphasizing the importance of initial embeddings in the post-enhancement unit for optimal performance in rumor detection.

Method	Pheme5		Pheme9	
	Acc	F1	Acc	F1
-w/o Neighbor	0.864	0.853	0.848	0.841
-w/o Attention	0.869	0.851	0.837	0.825
-w/o PE	0.851	0.833	0.848	0.841
-w/o Dynamic	0.858	0.845	0.853	0.850
DNKGAT	**0.892**	**0.879**	**0.894**	**0.888**

Table 4: Results of comparison among different variants on Pheme5 and Pheme9 Datasets.

4.5 Ablation Study

We investigate the effects of our proposed components by defining the following variations: (1)w/o neighbor: removing neighbor set between adjacent graphs; (2)w/o attention: removing adaptively adjust the weights of information with fixed weights; (3)w/o PE: replacing the post-enhancement unit with the concatenate operation; (4)w/o dynamic: utilizing static KGATs instead of dynamic KGATs. Specifically, we only use the knowledge graph representations in the final time stage as input, and the model only contains one DNKGAT unit.



(a) Pheme5 dataset (b) Pheme9 dataset (c) Pheme5 dataset (d) Pheme9 dataset

Figure 3: Early rumor detection accuracy with the increase of observation time or percentage of the number of comments.

From Table 4, it is evident that all ablation variants underperform compared to the complete model, with accuracy drops of 2.8% on Pheme5 and 4.6% on Pheme9 when the neighbor relevant set is removed, indicating the importance of neighboring information. Removing attention results in a 2.3% decrease on Pheme5 and a 5.7% decrease on Pheme9. Replacing the post-enhancement unit degrades accuracy by 4.1% on Pheme5 and 4.6% on Pheme9, while replacing the dynamic graph leads to a 3.4% decrease on Pheme5 and a 4.1% decrease on Pheme9. These results demonstrate the necessity of temporal neighboring information for better performance.

4.6 Early Rumor Detection Performance

Early rumor detection prevents widespread dissemination. We evaluate our method using two delay types: time since source post release and comment count. We compare against DDGCN and BiGCN, a Graph Convolutional Network (GCN) approach for rumor detection leveraging bidirectional propagation mechanisms—both propagation and diffusion structures—along with the textual content of posts. Figure 3 reveals that models struggle with few responsive posts due to lacking spatial and temporal structure. Our method, however, achieves high accuracy early on and consistently outperforms others, demonstrating the effectiveness of aggregating neighboring and knowledge information for early detection.

4.7 Case Study

To intuitively demonstrate the propagation in DNKGAT, we randomly sample source news and its comments, offering explanations. Figure 4 shows the visualization of adjacent nodes' connectivity. The propagation paths can be viewed as the evidence why the news is fake. As we can see, the connectivity $s-c2-TheScene(miniseries)$ has

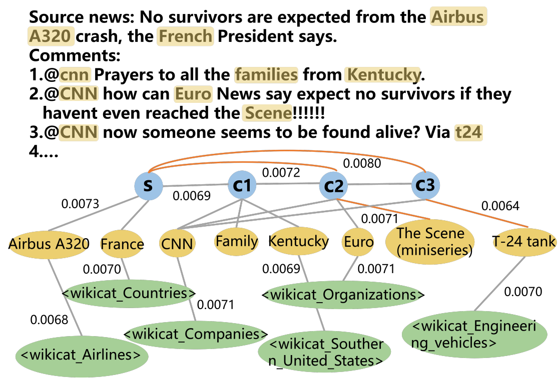


Figure 4: Real Example from Pheme9 dataset.

a higher attention score, labeled with the orange line. Hence, we can generate the explanation as *Euro News haven't reached the Scene, so they can't say "no survivors" which means the news is fake*. And the connectivity $s-c3-T24Tank$ shows *via t24, someone seems to be found alive which contradicts to "no survivors" in the source news*. Based on the above analysis, we can conclude that the source news is fake.

5 Conclusion

In this paper, we propose a dynamic neighbor-enhanced knowledge graph attention network (DNKGAT) to model the neighboring spatial structure, neighboring temporal structure, external knowledge and text information in one unified framework. DNKGAT concludes dynamic KGATs to capture dynamic neighboring knowledge information in propagation. Furthermore, we propose the post-enhancement unit to enhance the knowledge information in the end, in order to aggregate the source post information for classification incrementally. Experiments on two public datasets show that DNKGAT performs better than a set of strong baselines and supports rumor early detection.

6 Limitations

In this study, although the proposed DNKGAT method shows good performance in rumor detection, it still has some limitations. Firstly, the construction of the dynamic knowledge graph depends on external knowledge sources such as YAGO. The quality and comprehensiveness of these knowledge sources may affect the performance of the model. If there are errors or omissions in the knowledge graph, it may lead to inaccurate rumor detection results. Secondly, when dealing with the neighbor set in the dynamic neighbor-enhanced knowledge graph attention network, although sampling a fixed-size set of neighbors can reduce the computation overhead, it may also lose some useful information. In addition, the current model does not fully consider the potential influence of the interaction between different events on rumor detection. Each event is processed relatively independently, and the possible connections and mutual influences between events are not deeply explored. Finally, similar to some other studies, the evaluation of our method is mainly based on the existing public datasets Pheme5 and Pheme9. These datasets may have certain limitations in representing the real and complex social media environment, which may affect the generalization ability of the model to some extent.

References

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Stance detection with bidirectional conditional encoding. *arXiv: Computation and Language,arXiv: Computation and Language*.

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. *ArXiv*, abs/2001.06362.

Yi-Chin Chen, Zhao-Yang Liu, and Hung-Yu Kao. 2017. Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*.

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.

Limeng Cui, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, and Dongwon Lee. 2020. Deterrent: Knowledge guided graph attention network for detecting healthcare misinformation. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (SIGKDD)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Yaqian Dun, Kefei Tu, Chen Chen, Chunyan Hou, and Xiaojie Yuan. 2021. Kan: Knowledge-aware attention network for fake news detection. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Shuzhi Gong, Richard O. Sinnott, Jianzhong Qi, and Cécile Paris. 2023. Fake news detection through temporally evolving user interactions. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*.

Xiaobo Guo, Wenfang Lin, Youru Li, Zhongyi Liu, Lin Yang, Shuliang Zhao, and Zhenfeng Zhu. 2020. Dken: Deep knowledge-enhanced network for recommender systems. *Information Sciences*.

Zhiwei Guo, K. Yu, Alireza Jolfaei, Gang Li, Feng Ding, and Amin Beheshti. 2023. Mixed graph neural network-based fake news detection for sustainable vehicular social networks. *IEEE Transactions on Intelligent Transportation Systems*, 24:15486–15498.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Neural Information Processing Systems (NeurIPS)*.

Zhenyu He, Ce Li, Fan Zhou, and Yi Yang. 2021. Rumor detection on social media with event augmentations. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*.

Knut Hinkelmann, Sajjad Ahmed, and Flávio Corradini. 2022. Combining machine learning with knowledge engineering to detect fake news in social networks - a survey. *ArXiv*, abs/2201.08032.

Linmei Hu, Tianchi Yang, Luhao Zhang, Wanjun Zhong, Duyu Tang, Chuan Shi, Nan Duan, and Ming Zhou. 2021. Compare to the knowledge: Graph neural fake news detection with external knowledge. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Qi Huang, Chuan Zhou, Jia Wu, Mingwen Wang, and Bin Wang. 2019. Deep structure learning for rumor detection on twitter. *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Thomas Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *ArXiv*, abs/1609.02907.

Xiao-Yang Liu, Chenxiang Miao, Giacomo Fiumara, and Pasquale De Meo. 2024. Information propagation prediction based on spatial–temporal attention and heterogeneous graph convolutional networks.

IEEE Transactions on Computational Social Systems, 11:945–958.

Yang Liu and Yi fang Brook Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Jing Ma and Wei Gao. 2020. Debunking rumors on twitter with tree transformer. In *International Conference on Computational Linguistics (Coling)*.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard Jim Jansen, Kam-Fai Wong, and M. Cha. 2016. Detecting rumors from microblogs with recurrent neural networks. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

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. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (CIKM)*.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018a. Rumor detection on twitter with tree-structured recursive neural networks. In *Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018b. Rumor detection on Twitter with tree-structured recursive neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1980–1989, Melbourne, Australia. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, abs/1912.01703.

Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. Csi: A hybrid deep model for fake news detection. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM)*.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*.

Chenguang Song, Kai Shu, and Bin Wu. 2021a. Temporally evolving graph neural network for fake news detection. *Inf. Process. Manag.*, 58:102712.

Chenguang Song, Yiyang Teng, and Bin Wu. 2021b. Dynamic graph neural network for fake news detection. *2021 IEEE 7th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pages 27–31.

Mengzhu Sun, Xi Zhang, Jiaqi Zheng, and Guixiang Ma. 2022a. Ddgcn: Dual dynamic graph convolutional networks for rumor detection on social media. In *AAAI Conference on Artificial Intelligence (AAAI)*.

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

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio', and Yoshua Bengio. 2017. Graph attention networks. *ArXiv*, abs/1710.10903.

Hongwei Wang, Fuzheng Zhang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2019a. Multi-task feature learning for knowledge graph enhanced recommendation. *The World Wide Web Conference (WWW)*.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019b. Kgat: Knowledge graph attention network for recommendation. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (SIGKDD)*.

Youze Wang, Shengsheng Qian, Jun Hu, Quan Fang, and Changsheng Xu. 2020. Fake news detection via knowledge-driven multimodal graph convolutional networks. *Proceedings of the 2020 International Conference on Multimedia Retrieval (ICMR)*.

Huaiwen Zhang, Quan Fang, Shengsheng Qian, and Changsheng Xu. 2019. Multi-modal knowledge-aware event memory network for social media rumor detection. *Proceedings of the 27th ACM International Conference on Multimedia (MM)*.

Kaimin Zhou, Chang Shu, Binyang Li, and Jey Han Lau. 2019. Early rumour detection. In *Proceedings of the 2019 Conference of the North*.