LADKG: LLM-Augmented Dynamic Knowledge Graph for Fake News Detection

Anonymous ACL submission

Abstract

Fake news spreads rapidly on social media and causes serious societal harm. Existing methods rely on propagation structures, temporal signals, or external knowledge, but often model them separately and fail to capture dynamic diffusion, evolving comment-based knowledge, and high-order semantic relationships. We propose llm-augmented dynamic knowledge graph (LADKG), a unified framework for fake news detection. LADKG constructs a dynamic *post–entity–concept* knowledge graph from posts and user comments using large language models and updates it over time to capture semantic evolution. A multi-hop graph attention mechanism aggregates high-order neighborhood information for deep semantic reasoning. LADKG further introduces a post-enhancement unit to model fine-grained interactions between textual and knowledge representations. Experiments on two datasets show that LADKG consistently outperforms strong baselines, with notable gains in early-stage fake news detection. Our implementation is available at https://anonymous.4open.science/status/LADKG01-4B47.

1 Introduction

Social media platforms such as Twitter and Facebook have become primary channels for news dissemination and public discourse, enabling information to spread globally within seconds. However, limited content regulation has allowed fake news to proliferate rapidly (Allcott and Gentzkow, 2017). Fake news refers to intentionally false or misleading information, which can distort public understanding and lead to serious societal consequences, including the erosion of institutional trust and the escalation of social conflicts (Sharma et al., 2019). As a result, automatic fake news detection has become a critical research problem in natural language processing and social media analysis (Shu et al., 2017; Thorne and Vlachos, 2018). Existing approaches to fake news detection can be broadly grouped into three research directions: external knowledge integration, information propagation modeling, and temporal dynamics analysis.

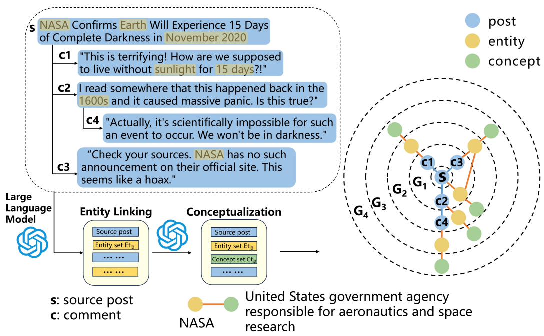


Figure 1: Evolution of the dynamic knowledge graph for fake news detection. Dotted circles represent the propagation of information over time, and based on the time intervals, we generate dynamic knowledge graphs (i.e. G_1, G_2, G_3 etc.).

Firstly, external knowledge bases (KBs) are integrated, such as Wikidata, to improve factual awareness and verification capability (Whitehouse et al., 2023). More recent studies leverage large language models (LLMs) to generate dynamic background knowledge and perform reasoning-based fact verification (Li et al., 2024; Chen et al., 2025), demonstrating strong potential in handling complex claims. Yet, they rely heavily on static or post-centric information and often *overlook the evolving background knowledge expressed in user comments,* which may contain corrections, supplementary evidence, and collective reasoning that accumulate over time.

Secondly, propagation-based approaches model the diffusion of misinformation through user interactions, which naturally form tree- or graph-structured patterns (Ma et al., 2018; Sun et al., 2022b). Graph neural networks are widely adopted to capture spatial diffusion structures and high-order relational dependencies among users and posts (Yin et al., 2024). However, propagation-based methods primarily focus on structural diffusion patterns while *under-modeling semantic evolution during information spread.*

Thirdly, temporal modeling approaches emphasize tracking the evolution of news over time, particularly for early-stage fake news detection, often using Transformer-based architectures to model temporal dynamics and distribution shifts (Hu J, 2025; Hu et al., 2023). *Such temporal dynamics are frequently modeled in isolation, without being tightly coupled with spatial propagation structures or external knowledge signals.*

To address these limitations, we propose llm-augmented dynamic knowledge graph (LADKG), a unified framework that integrates spatial and temporal propagation modeling with dynamic external knowledge reasoning. As illustrated in Figure 1, LADKG constructs a dynamic *post–entity–concept* tripartite knowledge graph by extracting entities, concepts, and relations from both source posts and user comments using a LLM, specifically Qwen-turbo. The knowledge graph is updated incrementally across multiple temporal stages of the news lifecycle, which allows the model to capture semantic evolution as the news unfolds. To capture deeper relational dependencies, LADKG employs a multi-hop graph attention mechanism that aggregates high-order neighborhood information beyond direct node connections.

A central component of LADKG is the post-enhancement unit, which explicitly couples textual representations with knowledge graph features. Instead of using simple feature concatenation or late fusion strategies, this unit models fine-grained interactions between post text embeddings and knowledge-aware representations. This design alleviates the decoupling of text and knowledge observed in previous work and leads to richer semantic representations. Finally, LADKG integrates contextual and knowledge-enhanced features for fake news classification.

The main contributions of this work are summarized as follows:

- We propose a unified framework for fake news detection that jointly models spatial–temporal propagation, evolving external knowledge from posts and user comments, and high-order semantic relationships.
- We construct a dynamic post–entity–concept tripartite knowledge graph and design a multi-hop graph attention mechanism with a post-enhancement unit to enable effective interaction between textual representations and knowledge graph features.
- Extensive experiments on two datasets show that LADKG consistently outperforms strong text-based, graph-based, and LLM-based baselines, especially in early-stage fake news detection.

2 Related Work

Related works of fake news detection are summarized as the following three research directions:

Spatial structure based fake news detection: Early work modeled message propagation trees using recursive neural networks (Ma et al., 2018). Recent developments utilize advanced GNNs to model evolving neighborhood patterns. For instance, Bidirectional Temporal-delay GCN (BTGCN-FND) captures dynamic spatial interactions in propagation graphs (Yin et al., 2024). Spatial structure modeling remains vital for capturing how misinformation spreads through network topology.

Temporal dynamics based fake news detection: Temporal modeling has advanced rapidly in recent years. Hu J (2025) propose dynamic temporal networks and Transformer encoders to detect fake news in early stages. Hu et al. (2023) introduce FTT, forecasting temporal trends to handle distribution shifts between training and test time. Logic-based frameworks like NDTL model diffusion as temporal logic to capture propagation patterns (Fionda, 2025). These works demonstrate the importance of capturing when and in what order messages propagate.

Knowledge graph and LLM-based fake news detection: Knowledge-graph-based methods enrich message content with external factual context. For instance, KMGCN (Wang et al., 2020) and CompareNet (Hu et al., 2021) leverage entity-concept structures. Recent work integrates LLMs to further enhance reasoning: LEKD (Chen et al., 2025) combines LLM retrieval with structured knowledge graphs; Re-Search (Li et al., 2024) employs multi-round retrieval-augmented LLM for claim verification; DKFND (Liu et al., 2025) enhances few-shot detection using dual-perspective LLM reasoning.

In summary, most existing approaches treat external knowledge, spatial diffusion, and temporal evolution as separate components, limiting their ability to jointly capture dynamic propagation behaviors, evolving comment-based knowledge, and high-order semantic relationships that are inherently intertwined in real-world fake news dissemination. Our method uniquely constructs and dynamically updates a knowledge graph from posts, jointly models spatial-temporal propagation through multi-hop attention, and deeply fuses textual and knowledge features.

3 Methodology

3.1 Problem Definition and Overview

The task of fake news detection can be defined as a binary classification problem aimed at identifying false or misleading information, such as fabricated news articles, social media posts, or other digital content. A news consists of the source post and comment contents as $o = \{(s, t_0), (c_1, t_1), \ldots, (c_{m-1}, t_{m-1})\}$, where s represents the source post released at time $t_0 = 0$. And c_i denotes the i-th comment in the sequence and t_i is its relative release time. Here, m indicates the total number of source post and comments. Comments are segmented into γ stages based on the duration, where γ is a hyperparameter. The duration of each stage is $\Delta t = \frac{t_{m-1}}{\gamma}$.

The goal is to learn a classification model f that maps news to a predefined category in $\mathcal{Y} = \{0, 1\}$, where $y = 0$ denotes true news and $y = 1$ denotes fake news.

The overall architecture of our fake news detection framework is illustrated in Figure 2. Given a source post and its comments, they are fed into dynamic knowledge graph construction module. It extracts external knowledge (entities and concepts) from this text using a LLM and incrementally constructs a temporal, tripartite knowledge graph, thereby capturing the evolving semantic relationships throughout the news lifecycle. The dynamic graph is fed into the dynamic knowledge graph attention. This module employs a multi-hop graph attention mechanism over the graph to learn comprehensive node representations that jointly encode spatial structure, temporal evolution, and knowledge semantics, enabling deep contextual reasoning. Then the learned knowledge-aware node representations are passed to the post-enhancement unit, explicitly models fine-grained interactions between the post textual and the derived knowledge-based representations, generating enriched, hybrid features that fuse both information streams. Finally, these enhanced features are aggregated and fed into the fake news classification module, which synthesizes them to make the final prediction on the veracity of the news.

3.2 Dynamic Knowledge Graph Construction

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 ambiguity, we integrate external knowledge from both source posts and their comments, which evolves dynamically as the news progresses.

Specifically, we employ Qwen-turbo to identify entities, extract concepts, and capture relationships through carefully designed prompts. This approach first extracts entities from the post, then dynamically leverages the rich semantic understanding capabilities of large language models (LLMs) to perform comprehensive semantic annotation on the entities. For instance, for the entities identified in the example post involving "Tesla" and "Model S", we obtain the corresponding concepts and relations: *Concept*(*Tesla*) = {Vehicle Manufacturer, Technology Company}, *Concept*(*Model S*) = {Electric Car, Automobile}, and *Relation*(*Tesla-Model S*) = {Manufactures, Produces}.

As mentioned above, we segment comments into γ stages based on their time duration. For a post and the first i stages of comments, we construct a corresponding dynamic knowledge graph G_i. The $(i+1)$-th stage of comments are then utilized to expand G_i to obtain G_{i+1}. Thus a set of dynamic knowledge graphs $\{G_1, \ldots, G_\gamma\}$ can be formed, where $G_i = \langle V_i, E_i \rangle$ is a post-entity-concept tripartite graph. V_i denotes the vertices and E_i denotes the edges. 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 weights are determined by the term frequency-inverse document frequency (TF-IDF) of the entity within

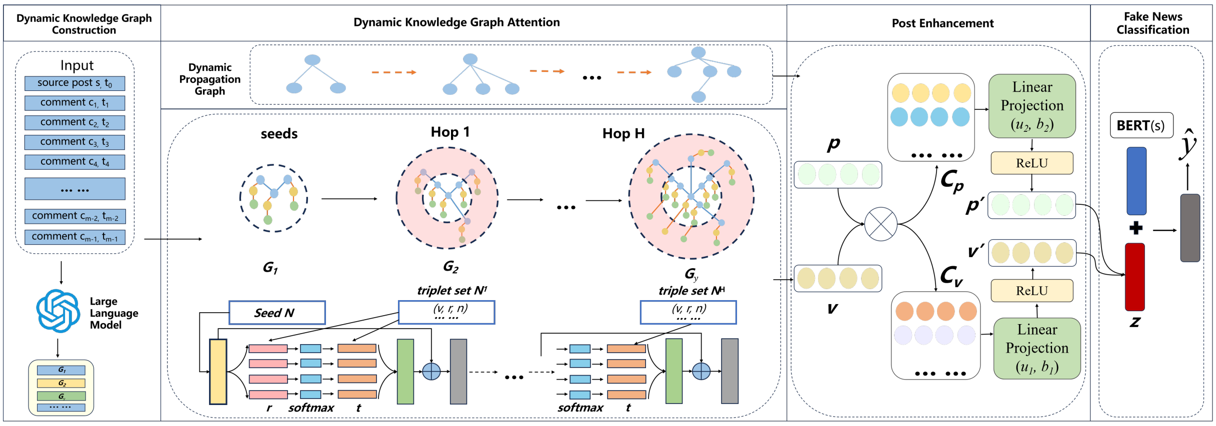


Figure 2: The framework of the proposed method. It consists of four components: (1) the leftmost block: input; (2) the left block: dynamic knowledge graph attention; (3) the right block: post-enhancement; (4) the rightmost block: fake news classification.

the post.

- **Entity-concept edges:** The relationships between entities and their corresponding concepts are quantified using Pointwise Mutual Information (PMI), which is calculated over a fixed-size sliding window of a global corpus.
- **Entity-entity edges:** These edges capture semantic relationships between different entities, which are generated by LLM through structured prompts.

3.3 Dynamic Knowledge Graph Attention

This module generates node representations that incorporate spatial, temporal, and knowledge information. By leveraging the graph attention network architecture (Velickovic et al., 2017), we compute attentive weights for cascaded information propagation to reveal the importance of different node connections. We first describe the structure of a single layer, which consists of information propagation and aggregation mechanisms, and then extend this framework to multiple layers for enhanced representation learning.

Information Propagation: Entities enrich posts by propagating aggregated information from multiple associated concepts. To characterize nodes' hierarchically extended propagation in terms of KG, we recursively define the set of k-hop neighbor set and triplet set for $G_i = \langle V_i, E_i \rangle$:

Definition 1 (Neighbor Set) *Given the knowledge graph G_i in i-th stage, the k-hop neighbor set for graph G_i is defined as:*

$$\mathcal{N}_i^k = \{n \mid (v, r, n) \in G_i \text{ and } v \in \mathcal{N}_i^{k-1}\} \\ k = 1, 2, 3, ..., K.$$

where v denotes the head entity, r is the relation, n is the tail entity. K is the largest hop number. $\mathcal{N}_i^0 = V_i$ is the set of vertices at the beginning of i-th stage, which can be seen as the seed set of the knowledge graph.

Definition 2 (Triplet Set) *The k-hop triplet set of knowledge graph G_i is defined as the set starting from $\mathcal{N}_i^{k-1}$:*

$$\mathcal{T}_i^k = \{(v, r, n) \mid (v, r, n) \in G_i \text{ and } v \in \mathcal{N}_i^{k-1}\} \\ k = 1, 2, 3, ..., K.$$

where K is the largest hop number.

For the knowledge graph G_i at the i-th stage, it already contains the nodes from G_{i-1}. Then for the newly added nodes, we use BERT (Devlin et al., 2019) as the encoder, each head node is encoded into an embedding $\boldsymbol{v}_i$. Each triple (v, r, n) is represented by $(\boldsymbol{v}_i, \boldsymbol{r}_i, \boldsymbol{n}_i)$. Given a node v as the head node and the one-hop triplet set $\mathcal{T}_i^1$, a relevance probability is assigned to each triple (v, r, n) by comparing the head node embedding $\boldsymbol{v}_i$ with its neighbor node embedding $\boldsymbol{n}_i$ and the corresponding relation embedding $\boldsymbol{r}_i$:

$$p_i = \frac{\exp\left(\boldsymbol{v}_i^\top \cdot (\boldsymbol{r}_i \odot \boldsymbol{n}_i)\right)}{\sum_{(v_i, r_i', n_i') \in \mathcal{T}_i^1} \exp\left(\boldsymbol{v}_i^\top \cdot (\boldsymbol{r}_i' \odot \boldsymbol{n}_i')\right)} \quad (1)$$

where $\odot$ is the element-wise multiplication and $\cdot$ denotes the inner product. $(v_i, r_i', n_i') \in \mathcal{T}_i^1$ indicates all the triplets with the head node as v in the triplet set $\mathcal{T}_i^1$.

The relevance probability p_i controls the decay factor on each propagation edge $(\boldsymbol{v}_i, \boldsymbol{r}_i, \boldsymbol{n}_i)$, indicating how much information is being propagated from $\boldsymbol{n}_i$ to $\boldsymbol{v}_i$ conditioned on relation $\boldsymbol{r}_i$, regarded as the similarity of the head entity $\mathbf{v}_i$ and the tail entity $\boldsymbol{n}_i$ measured in the space of relation $\boldsymbol{r}_i$. 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 then compute a weighted neighbor representation for the given head node $\boldsymbol{v}_i$:

$$\boldsymbol{n}_i^{(1)} = \sum_{(v_i, r_i, n_i) \in \mathcal{T}_i^1} p_i \boldsymbol{n}_i, \tag{2}$$

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 $\boldsymbol{r}$ between $\boldsymbol{v}$ and $\boldsymbol{n}$, encoding more information during propagation.

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

$$\begin{aligned} f(\boldsymbol{v}_i, \boldsymbol{n}_i^{(1)}) =& \mathrm{LeakyReLU}(\mathbf{W_1}(\boldsymbol{v}_i + \boldsymbol{n}_i^{(1)})) + \\ & \mathrm{LeakyReLU}(\mathbf{W_2}(\boldsymbol{v}_i \odot \boldsymbol{n}_i^{(1)})), \end{aligned}$$

where $\mathbf{W_1}, \mathbf{W_2}$ are trainable weight matrices, and $\odot$ denotes the element-wise product. Distinct from GCN and GraphSage aggregators, we additionally encode the feature interaction between $\boldsymbol{v}_i$ and $\boldsymbol{n}_i^{(1)}$. This term makes the information being propagated sensitive to the affinity between $\boldsymbol{v}_i$ and $\boldsymbol{n}_i^{(1)}$, 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 hop, we recursively formulate the representation of KG as:

$$\boldsymbol{v}_i^{(l)} = f(\boldsymbol{v}_i^{(l-1)}, \boldsymbol{n}_i^{(l-1)}) \tag{3}$$

After K propagation layers, we obtain the final representation $\boldsymbol{v}_\gamma^{(K)}$, where K is the maximum hop number and γ denotes the number of stages into which comments are segmented by time.

3.4 Post-Enhancement Unit

Motivated by the idea of root feature enhancement in (Bian et al., 2020), we emphasize the pivotal role of the source post in fake news detection. To better integrate the original post features with knowledge-enhanced representations, we design a cross-information sharing mechanism inspired by multi-task fusion (Wang et al., 2019a) and dual-channel interaction approaches (Guo et al., 2020).

Specifically, we concatenate all the comments' initial text and encode into an embedding $\boldsymbol{p}$. With the latent representation $\boldsymbol{v}_\gamma^{(K)}$ produced by the dynamic knowledge graph attention. This fusion enables incremental learning by capturing both the raw semantic content and context-enriched representations. To model fine-grained interactions between posts and knowledge nodes, we compute two cross-feature matrices using pairwise outer products:

$$\mathbf{C_v} = [\boldsymbol{v}_\gamma^{(K)}]\boldsymbol{p}^\top, \quad \mathbf{C_p} = \boldsymbol{p}[\boldsymbol{v}_\gamma^{(K)}]^\top, \tag{4}$$

where $\boldsymbol{v}_\gamma^{(K)}, \boldsymbol{p} \in \mathbb{R}^d$ are the embeddings of knowledge graph nodes and post nodes, respectively. Here, $\mathbf{C_v}$ reflects how knowledge influences the post semantics, while $\mathbf{C_p}$ models the influence from the post to the knowledge representation.

We project the cross-feature matrices back to the latent space via linear compression:

$$\boldsymbol{v'} = \mathrm{ReLU}(\mathbf{C_v}\boldsymbol{u_1} + \boldsymbol{b_1}) \tag{5}$$
$$\boldsymbol{p'} = \mathrm{ReLU}(\mathbf{C_p}\boldsymbol{u_2} + \boldsymbol{b_2}) \tag{6}$$

where $\boldsymbol{u_1}, \boldsymbol{u_2} \in \mathbb{R}^{d\times 1}$ and $\boldsymbol{b_1}, \boldsymbol{b_2} \in \mathbb{R}^d$ are trainable parameters. The resulting feature vectors $\boldsymbol{v'}$ and $\boldsymbol{p'}$ represent cross-modal feature interactions and semantic alignment.

Through this post-enhancement unit, our framework effectively integrates knowledge-aware context with original textual information, producing enriched semantics for improved classification.

3.5 Fake News Classification

This component performs final fake news classification by aggregating semantic and knowledge-enhanced representations. It uses a series of fully connected layers with a sigmoid activation function to predict whether the news is fake or true. To obtain a unified representation for the news, we apply mean pooling over node embeddings from the post-enhancement module:

$$\boldsymbol{z} = \mathrm{MeanPooling}(\{\boldsymbol{v'}, \boldsymbol{p'}\}) \tag{7}$$

We then concatenate $\boldsymbol{z}$ with contextual features of the source post s extracted by a pretrained BERT encoder (Devlin et al., 2019), and apply a linear transformation followed by Leaky ReLU:

$$\tilde{\boldsymbol{s}} = \mathrm{LeakyReLU}(\mathrm{Linear}(\boldsymbol{z}||\mathrm{BERT}(s))) \quad (8)$$

The final classification prediction $\hat{y}$ is computed as:

$$\hat{y} = \sigma(\mathbf{W_f}\tilde{s} + \mathbf{b_f}) \quad (9)$$

where $\mathbf{W_f}$ and $\mathbf{b_f}$ are trainable weight and bias parameters.

The model is optimized using the binary cross-entropy loss, which penalizes the divergence between predicted probabilities and ground truth labels:

$$\mathcal{L}_c = -\sum_i y_i \log \hat{y}_i \quad (10)$$

where $y_i \in \{0, 1\}$ is the true label for instance i.

4 Experiments

4.1 Datasets

We evaluate LADKG on Pheme5 (Kochkina et al., 2018) and Pheme9 (Zubiaga et al., 2017). Each dataset contains a collection of social media events labeled as either fake news or real news. Pheme5 consists of events from five high-impact news topics: Charliehebdo, Ferguson, Germanwings-crash, Otawashooting, and Sydney-siege. Each event includes a source post, its replies, corresponding timestamps, and propagation structures. Pheme9 extends Pheme5 by incorporating four additional events: Ebola-Essien, Gurlitt, PrinceToronto, and Putinmissing, maintaining the same structural format. Dataset statistics are in Table 3. Detailed statistics are provided in Appendix.

4.2 Comparison Methods

We compare with the following baselines: **BiGCN** (Bian et al., 2020): A GCN approach using bidirectional propagation and post textual content. **DDGCN** (Sun et al., 2022a): A dual dynamic GCN that learns message propagation and knowledge graph dynamics. **ASTKN** (Li et al., 2023): Uses dynamic graph attention networks with a novel attention mechanism to merge propagation and knowledge structure data. **NRA MOS-GAT** (Patel et al., 2024): Leverages BERTweet for textual encoding and GAT for graph-based propagation modeling, enhanced by a multi-oversampling strategy to mitigate class imbalance and boost early-stage fake news detection. **NRA MOS-GCN** (Patel et al., 2024): Similar to NRA MOS-GAT but employs GCN for structural aggregation, providing competitive results with improved robustness on PHEME datasets. **MRAN** (Yang et al., 2024): A multimodal relationship-aware attention network that extracts hierarchical semantic features from text and visual cues to model intra- and inter-modal dependencies.

4.3 Experiment Setup

We follow standard experimental protocols as adopted in prior works. The implementation details, training settings, and evaluation protocol are described in Appendix.

4.4 Performance Analysis

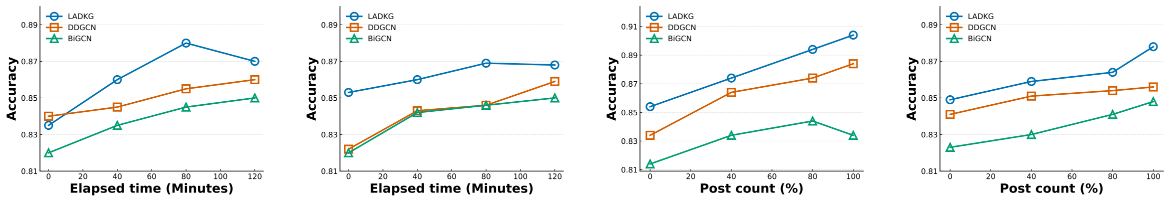
Table 1 presents the performance of all compared models. Our model consistently achieves the best results across all metrics on both Pheme5 and Pheme9 datasets, confirming that incorporating dynamic neighbor information significantly benefits fake news detection. Traditional models such as SVM-BOW perform poorly due to their dependence on handcrafted features, which fail to capture complex semantic patterns. In contrast, deep learning-based methods like CNN, BiLSTM, and BERT demonstrate substantial improvements, with BERT showing particular strength in modeling semantic representations.

While BERT is competitive among text-only approaches, graph-based models such as Bi-GCN further enhance performance, highlighting the importance of integrating structural information from propagation graphs with textual content. DDGCN and ASTKN, which incorporate dynamic propagation and knowledge structures, surpass most prior baselines, indicating that knowledge-aware modeling provides complementary advantages. MOS-GAT and MOS-GCN use BERTweet plus GAT/GCN propagation with multi-oversampling to counter class imbalance; their moderate results signal that oversampling alone cannot compensate for the lack of knowledge or neighbor cues. MRAN deploys multimodal relation-aware attention for text–visual interplay, yet without explicit propagation and knowledge modeling its gains remain limited.

Our method achieves the best overall performance by jointly modeling temporal dynamics, propagation structures, and external knowledge from both source posts and user comments. Additionally, it adaptively modulates node influence

Method	Features					Pheme5			Pheme9		
	Text	Temporal	Spatial	KG	Neighboring	Acc	Rec	F1	Acc	Rec	F1
SVM-BOW	✓					0.669	0.524	0.529	0.688	0.512	0.515
BERT	✓					0.815	0.779	0.796	0.821	0.788	0.803
CNN	✓					0.787	0.702	0.719	0.795	0.673	0.701
BiLSTM	✓					0.795	0.691	0.725	0.794	0.677	0.701
TD-RvNN	✓		✓			0.821	0.764	0.769	0.804	0.803	0.803
BU-RvNN	✓		✓			0.817	0.761	0.762	0.789	0.788	0.788
Bi-GCN	✓		✓			0.829	0.814	0.818	0.847	0.834	0.835
DDGCN	✓	✓	✓	✓		0.844	0.813	0.823	0.855	0.841	0.843
ASTKN	✓	✓	✓	✓		0.872	0.852	0.856	0.867	0.851	0.855
NRA MOS-GAT	✓	✓	✓			76.45	-	73.64	78.36	-	73.07
NRA MOS-GCN	✓	✓	✓			77.52	-	74.51	77.49	-	71.58
MRAN	✓					0.870	0.868	0.874	-	-	-
LADKG	✓	✓	✓	✓	✓	**0.899**	**0.893**	**0.889**	**0.876**	**0.852**	**0.859**

Table 1: Comparison of different models on the Pheme5 and Pheme9 datasets. Results are averaged over 5-fold cross-validation.



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

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

Method	Pheme5		Pheme9	
	ACC	F1	ACC	F1
-w/o Neighbor	0.873	0.856	0.860	0.850
-w/o Attention	0.847	0.825	0.830	0.821
-w/o PE	0.851	0.833	0.843	0.842
-w/o Dynamic	0.858	0.839	0.865	0.849
-w/o LLM	0.892	0.879	0.874	0.858
LADKG	**0.899**	**0.889**	**0.876**	**0.859**

Table 2: Performance of LADKG and its variants on Pheme5 and Pheme9 datasets.

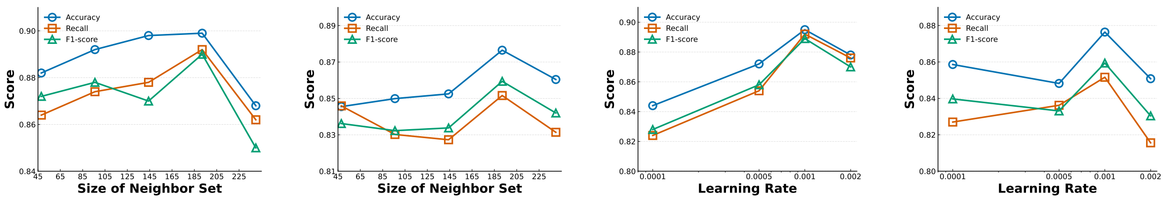
and emphasizes initial embeddings through a post-enhancement unit, leading to superior fake news classification accuracy.

4.5 Ablation Study

We conduct ablation experiments to evaluate the contribution of each core component in LADKG. The following variants are compared:

- **w/o Neighbor**: Removes the neighbor set between adjacent graphs to assess the effect of temporal entity relationships.
- **w/o Attention**: Replaces the adaptive attention mechanism with fixed uniform weights.
- **w/o PE**: Replaces the post-enhancement unit with a simple concatenation operation.
- **w/o Dynamic**: Uses only the final-stage knowledge graph representations and a single KGAT layer, discarding the dynamic modeling across time stages.
- **w/o LLM**: Builds the knowledge graph by extracting entities and concepts from the YAGO database instead of using LLM-generated nodes.

As shown in Table 2, all variants yield inferior performance compared to the full LADKG model. Specifically, removing neighboring information causes an accuracy drop of 2.6% on Pheme5 and 1.6% on Pheme9, indicating its importance. Eliminating the attention mechanism results in 5.2% and 4.6% drops on Pheme5 and Pheme9, respectively. Replacing the post-enhancement unit leads to a 4.8% drop on Pheme5 and 3.3% on Pheme9. Using a static graph (w/o Dynamic) decreases performance by 4.1% and 1.1%, respectively. Finally, using YAGO-derived entities and concepts instead of LLM-augmented ones—yields a 0.7% accuracy



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

Figure 4: Parameter study for LADKG.

decrease on Pheme5 and a 0.2% accuracy decrease on Pheme9. These results highlight the effectiveness of modeling temporal dynamics, neighbor-aware structures, LLM-augmented external knowledge and adaptive enhancement mechanisms in fake news detection.

4.6 Early Fake News Detection Performance

Early fake news detection targets the initial stage of news release, which is crucial to prevent widespread misinformation. We evaluate our method under two types: (1) elapsed time since the source post was released, and (2) the number of accumulated comments, comparing it with **DDGCN** and **BiGCN**. As shown in Figure 3, LADKG consistently outperforms baselines in early stages. This shows the effectiveness of the proposed method in more difficult scenario. A possible reason is that baselines have limited context. Yet LADKG introduces external knowledge, leveraging neighbor and knowledge information for timely detection.

4.7 Parameter Study

To evaluate LADKG's sensitivity to key configurations, we examine the effects of neighbor set size S (48–192) and learning rate lr (0.0001–0.002), keeping other parameters fixed. As shown in Figure 4, performance improves as S increases, peaking at 192, beyond which excessive neighbors introduce noise and reduce accuracy. For the learning rate, $lr = 0.001$ yields the best performance across Accuracy, Recall, and F1-score, reflecting a good balance between convergence and generalization.

4.8 Case Study

To intuitively demonstrate the propagation in LADKG, we randomly sample source news and its comments, offering explanations. Figure 5 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

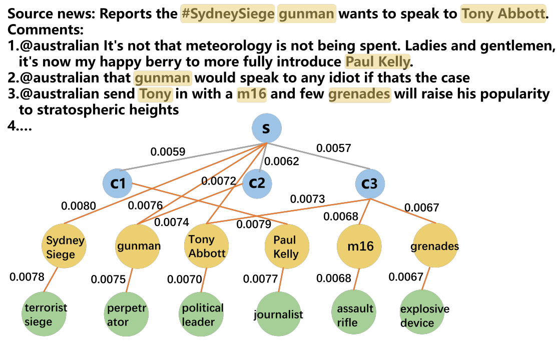


Figure 5: Real example from Pheme9 dataset.

connectivity $s - c2 - gunman$ has a higher attention score, indicating that comment2's mocking of the "gunman" claim that "he would speak to any idiot" directly undermines the source news. Likewise, the path $s - c3 - M16$ reveals comment3's sarcastic suggestion, which is absurd in the context of real negotiation. Together, these high-weight propagation paths serve as concrete evidence that the original claim is unfounded—hence, the source news is fake.

Conclusion

In this paper, we propose LADKG, a unified framework for fake news detection that integrates spatial propagation structure, temporal dynamics, external knowledge, and textual information. LADKG constructs dynamic post-entity-concept knowledge graphs with the help of large language model (LLM), and employs a multi-hop attention mechanism to capture evolving semantic relationships across high-order neighbors. Additionally, a post-enhancement unit is introduced to model fine-grained interactions between posts and knowledge nodes, enriching representations for more accurate classification. Experimental results on two real-world datasets demonstrate that LADKG consistently outperforms strong baselines, especially in early-stage fake news detection.

Limitations

In this study, although the proposed LADKG method shows good performance in fake news detection, it still has some limitations. Firstly, when dealing with the neighbor set in the dynamic knowledge graph attention, 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 news on fake news 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. 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

Hunt Allcott and Matthew Gentzkow. 2017. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–236.

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.

Xueqin Chen, Xiaoyu Huang, Qiang Gao, Li Huang, and Guisong Liu. 2025. Enhancing text-centric fake news detection via external knowledge distillation from llms. *Neural Networks*, 187:107377.

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)*.

Valeria Fionda. 2025. Logic-based analysis of fake news diffusion on social media. *Social Network Analysis and Mining*, 15.

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*.

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

Beizhe Hu, Qiang Sheng, Juan Cao, Yongchun Zhu, Danding Wang, Zhengjia Wang, and Zhiwei Jin. 2023. Learn over past, evolve for future: Forecasting temporal trends for fake news detection. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 116–125, Toronto, Canada. Association for Computational Linguistics.

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)*.

Li Z Hu J, Zhang J. 2025. Tracing truth: Dynamic temporal networks for multi-modal fake news detection. *PeerJ Computer Science 11:e2998*.

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

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task learning for rumour verification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3402–3413, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Guanghua Li, Wensheng Lu, Wei Zhang, Defu Lian, Kezhong Lu, Rui Mao, Kai Shu, and Hao Liao. 2024. Re-search for the truth: Multi-round retrieval-augmented large language models are strong fake news detectors. *Preprint*, arXiv:2403.09747.

Hui Li, Guimin Huang, Cheng Li, Jun Li, and Yabing Wang. 2023. Adaptive spatial–temporal and knowledge fusing for social media rumor detection. *Electronics*.

Ye Liu, Jiajun Zhu, Xukai Liu, Haoyu Tang, Yanghai Zhang, Kai Zhang, Xiaofang Zhou, and Enhong Chen. 2025. Detect, investigate, judge and determine: A knowledge-guided framework for few-shot fake news detection. *Preprint*, arXiv:2407.08952.

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

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.

Shaswat Patel, Prince Bansal, and Preeti Kaur. 2024. Rumour detection on benchmark twitter datasets using graph neural networks with data augmentation. *Social Network Analysis and Mining*, 14(1):163.

Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. 2019. Combating fake news: A survey on identification and mitigation techniques. *ACM transactions on intelligent systems and technology (TIST)*, 10(3):1–42.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, 19(1):22–36.

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)*.

James Thorne and Andreas Vlachos. 2018. Automated fact checking: Task formulations, methods and future directions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3346–3359.

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)*.

Chenxi Whitehouse, Tillman Weyde, Pranava Madhyastha, and Nikos Komninos. 2023. Evaluation of fake news detection with knowledge-enhanced language models. *Preprint*, arXiv:2204.00458.

Hongyu Yang, Jinjiao Zhang, Liang Zhang, Xiang Cheng, and Ze Hu. 2024. Mran: Multimodal relationship-aware attention network for fake news detection. *Computer Standards & Interfaces*, 89:103822.

Yunfei Yin, Zhiling Chen, and Xianjian Bao. 2024. Bidirectional temporal-delay graph convolutional network for detecting fake news. *Engineering Applications of Artificial Intelligence*, 133:108368.

Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2017. Exploiting context for rumour detection in social media. In *Social Informatics*.

A Appendix

Entity Relation Extraction Prompt

You are an information extraction system. Given a short social media post or sentence, extract the following:

1. Extraction Tasks

1. A list of unique named entities mentioned in the text, including people, organizations, locations, and abstract concepts such as events and dates.
2. A list of semantic triples in the form `[head_entity, relation, tail_entity]`, where each element is a real-world entity or concept explicitly or implicitly stated in the text.

2. Extraction Rules

- Each entity must appear in the original text or be an implied real-world concept (e.g., "President", "2022").
- Entities should be non-redundant and canonical (e.g., resolve pronouns like "he", "she", or "it" to the correct entity when possible).
- Relations must describe meaningful semantic connections (e.g., "is CEO of", "born in", "acquired").
- Do not include verbs as standalone entities.
- Output must be formatted as a valid JSON object.

3. Output Format

Output format: `{ "entity": [...], "relation": [[head, relation, tail], ...] }`

4. Example

Text: "Elon Musk is the CEO of Tesla and he acquired Twitter in 2022."
Output: `{ "entity": ["Elon Musk", "Tesla", "Twitter", "2022"], "relation": [ ["Elon Musk", "is CEO of", "Tesla"], ["Elon Musk", "acquired", "Twitter"], ["Elon Musk", "acquired Twitter in", "2022"] ] }`

Concept Typing Prompt

You are a knowledge abstraction system. Given a named entity, return its most relevant high-level concept category.

1. Tying Tasks

The concept should describe what type of thing the entity is (e.g., 'person', 'organization', 'event', 'location', 'disease', 'company', 'date', etc.).

2. Tying Rules

- Choose the most intuitive and commonly accepted category.
- Return only one lowercase word or short phrase.
- Do not add quotation marks, explanations, or extra formatting.
- Be consistent and concise.

3. Example

Input: COVID-19 → Output: disease
Input: UN → Output: organization
Input: Barack Obama → Output: person
Input: Tesla → Output: company
Input: 2022 → Output: date
Input: World War II → Output: event

Dataset statistics

Statistics	Pheme5	Pheme9
# of Posts	103,212	105,354
# of events	5,802	6,425
# of Non-rumors	3,830	4,023
# of Rumors	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 3: Statistics of Datasets

Experiment Setup

Following prior works, we adopt the default optimization settings as reported in their original papers. Our model is implemented using the PyTorch framework (Paszke et al., 2019), with parameters optimized via the Adam algorithm. We use BERT-base (Devlin et al., 2019) as the encoder for the source posts, initialized from pre-trained weights and fine-tuned on each dataset. The number of time stages is set to $\gamma = 3$, and the model is trained for 5 epochs. For both Pheme5 and Pheme9 datasets, we split the data into training, validation, and test sets with a 7:1:2 ratio. The best model is selected based on validation performance. To account for label imbalance across datasets, we adopt Accuracy (Acc), Recall (Rec), and F1-score as evaluation metrics. We further perform 5-fold cross-validation by randomly partitioning each dataset into five subsets and reporting the average results.