Structure-aware Propagation Generation with Large Language Models for Fake News Detection

Mengyang Chen[1,2]   **Lingwei Wei**[1*]   **Wei Zhou**[1]   **Songlin Hu**[1,2]

[1]Institute of Information Engineering, Chinese Academy of Sciences
[2]School of Cyber Security, University of Chinese Academy of Sciences
{chenmengyang, weilingwei, zhouwei, husonglin}@iie.ac.cn

Abstract

The spread of fake news on social media poses a serious threat to public trust and societal stability. While propagation-based methods improve fake news detection by modeling how information spreads, they often suffer from incomplete propagation data. Recent work leverages large language models (LLMs) to generate synthetic propagation, but typically overlooks the structural patterns of real-world discussions. In this paper, we propose a novel structure-aware synthetic propagation enhanced detection (StruSP) framework to fully capture structural dynamics from real propagation. It enables LLMs to generate realistic and structurally consistent propagation for better detection. StruSP explicitly aligns synthetic propagation with real-world propagation in both semantic and structural dimensions. Besides, we also design a new bidirectional evolutionary propagation (BEP) learning strategy to better align LLMs with structural patterns of propagation in the real world via structure-aware hybrid sampling and masked propagation modeling objective. Experiments on three public datasets demonstrate that StruSP significantly improves fake news detection performance in various practical detection scenarios. Further analysis indicates that BEP enables the LLM to generate more realistic and diverse propagation semantically and structurally.

1 Introduction

The rapid advancement of online media has led to an alarming rise in fake news, posing significant threats to public trust and societal stability (Fisher et al., 2016; Vosoughi et al., 2018; Faris et al., 2017).

Existing methods of fake news detection mainly focus on textual content such as news text and contexts (Castillo et al., 2011; Ma et al., 2015; Yu et al., 2017), and propagation information such as interactions between users (Lu and te Li, 2020; Su et al.,
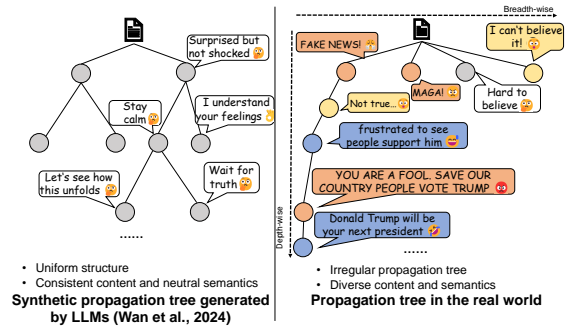

Figure 1: A comparison of LLM-generated propagation (e.g., DELL (Wan et al., 2024)) and original propagation of the news "*Donald Trump has been disqualified from running for president.* "

2023; Liu and fang Brook Wu, 2018; Bian et al., 2020; Wei et al., 2021; Wu and Hooi, 2023; Chen et al., 2024). Despite their promise, propagation-based methods often suffer in incomplete propagation scenarios due to limited data collection and some malicious user interactions on social media (Ma et al., 2022; Wei et al., 2024). Recently, LLMs have shown potential in alleviating data scarcity by simulating user-generated propagation through role-playing approaches (Wan et al., 2024; Nan et al., 2024; Liu et al., 2024; Qiu et al., 2025; Yue et al., 2024).

However, these LLM-enhanced methods typically operate only at the semantic level, neglecting the structural patterns of real-world propagation. As shown in Figure 1, the generated propagation trees often exhibit overly uniform structures due to the predefined branch probability (Wan et al., 2024) and prompt-induced alignment behavior (Denison et al., 2024; Sharma et al.), failing to capture the irregular branching and hierarchical depth that characterize real-world information spread (Zhao et al., 2020). In addition, their generated content often lacks emotional diversity and context sensitivity, tending toward overly cautious or generic tones (Muñoz-Ortiz et al., 2023; Frisch and Giulianelli,

---

*Corresponding author.

2024). The mismatch between the generated and real propagation structures and semantics significantly limits their effectiveness in downstream detection, particularly in early detection and cross-platform generalization. This highlights the need for a structure-aware generation framework that captures both the semantic plausibility and structural dynamics of real-world propagation.

To address the above limitation, we propose a novel **Stru**cture-aware **S**ynthetic **P**ropagation enhanced detection (StruSP) framework to fully capture sufficient features from real-world and LLM-generated synthetic propagation structure. StruSP enriches incomplete propagation trees by generating realistic and structurally consistent propagation paths, thereby improving the effectiveness of propagation-based fake news detection. To ensure the generated propagation reflects real-world structural dynamics, we introduce a bidirectional evolutionary propagation (BEP) learning strategy to align LLMs with structural patterns of propagation in the real world. BEP consists of two main components. The structure-aware hybrid sampling module first samples propagation substructures via both breadth-wise and depth-wise progression of available propagation trees. Based on these sampled paths, the masked propagation modeling objective captures structural dependencies by reconstructing masked nodes in both forward and backward directions. This design enables the LLM to effectively learn structural evolution patterns in real-world propagation, equipping it with the ability to enrich incomplete propagation through a structure propagation enhancement module for more accurate detection.

Experiments on three real-world datasets demonstrate that StruSP not only improves detection performance under incomplete propagation conditions but also generates propagation patterns that closely match real data, consistently surpassing baseline methods across structural and semantic metrics.

The contributions of this work can be summarized as follows:

1) We propose StruSP, a novel structure-aware framework for fake news detection in incomplete propagation scenarios. StruSP enhances fake news detection by generating realistic and structure-aware propagation trees that integrate both semantic and structural signals from partial real propagation.

2) To capture the structural evolution of the real propagation, we introduce a bidirectional evolu-

tionary propagation learning strategy. It enables LLMs to generate structurally diverse and coherent propagation trees.

3) We conduct extensive experiments on three real-world datasets, demonstrating that StruSP significantly improves detection performance and better aligns with real propagation in both structure and semantics.

2 Related Work

Fake News Detection The goal of detecting fake news is to identify and assess the authenticity of a piece of information. Existing methods for detecting fake news mainly focus on two aspects: textual content and propagation of news.

Content-based Fake News Detection Methods extract semantic patterns from news content for detection through feature engineering (Castillo et al., 2011; Popat, 2017; Ma et al., 2015) and a wide array of deep learning architectures, including neural networks (Ruchansky et al., 2017; Karimi and Tang, 2019) and pre-trained language models (Kaliyar et al., 2021; Jwa et al., 2019). Some works also integrate tasks such as stance detection and sentiment analysis with fake news detection, enabling multi-task learning (Luvembe et al., 2023; Hamed et al., 2023). Since some fake news creators imitate the style of real news, methods based solely on news content often face limitations. Consequently, some researchers use news comments as a basis for assessing the authenticity of news (Shu et al., 2019; Zhang et al., 2021).

Propagation-based Fake News Detection Methods capture the propagation patterns of news by modeling the interactions between news and comments into time series (Ma et al., 2016; Liu and fang Brook Wu, 2018) or topological structures such as propagation trees (Ma et al., 2018; Hu et al., 2021) and propagation graphs (Bian et al., 2020; Wei et al., 2021, 2022). Some studies further explore multi-relational interactions between the users and news in the propagation graph (Yuan et al., 2020; Dou et al., 2021). However, these methods suffer significant performance losses when confronted with scenarios of incomplete propagation (Wei et al., 2024; Ma et al., 2022).

LLM-based Propagation Generation LLMs have been proven to have the potential to simulate human behavior (Argyle et al., 2023) and possess a certain level of social knowledge (Choi et al., 2023).
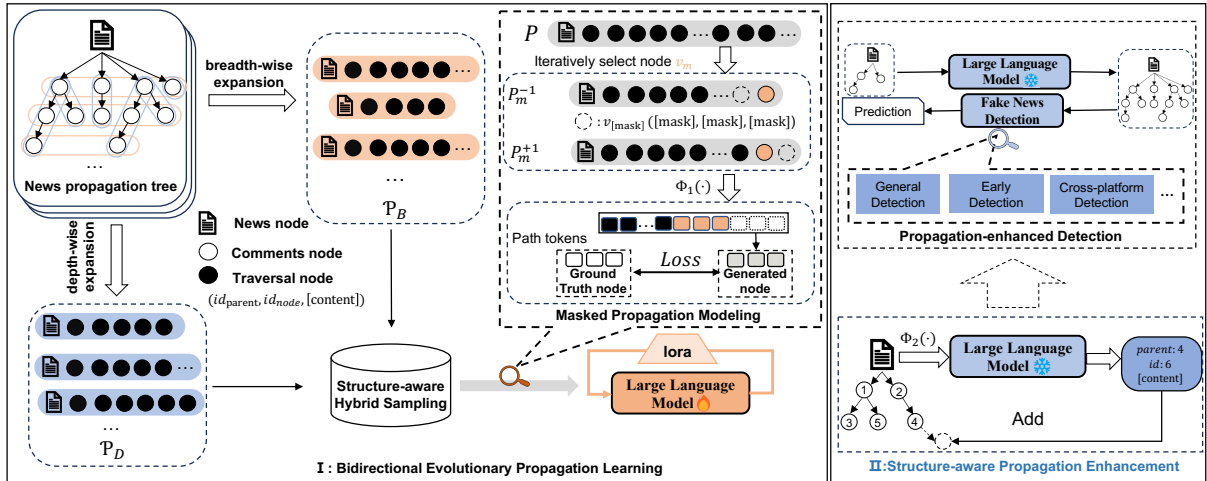
Figure 2: Overall framework of StruSP. We first perform bidirectional evolutionary propagation learning to capture the structural evolution of real propagation. We sample a set of propagation paths for both breadth-wise and depth-wise evolution based on the propagation tree, and then train the LLM to reconstruct masked nodes in forward and backward directions along the sampled path. Then, we generate synthetic propagation to enrich existing propagation for the downstream detection.

Recently, some studies have utilized LLMs to simulate social users and generate social contexts (Gao et al., 2023; Liu et al., 2024; Jiang and Ferrara, 2023). These methods typically leverage LLMs to role-play various users and generate replies to news items, thereby forming synthetic propagation data (Nan et al., 2024; Qiu et al., 2025; Wan et al., 2024). For instance, Nan et al. (2024) simulated discussions by prompting LLMs to adopt different user identities and respond iteratively. Qiu et al. (2025) further guided the generation process by modeling user behavior through a multilayer perceptron based on historical interactions. Wan et al. (2024) attempted to construct propagation structures by probabilistically controlling whether the LLM comments directly on the news or replies to existing comments.

However, existing methods either overlook the modeling of propagation structures or generate propagation patterns that do not match real-world structures. Our proposed StruSP framework explicitly models the structural evolution of propagation and produces propagation trees that better reflect real-world topologies, improving fake news detection performance, especially in scenarios with incomplete propagation.

3 StruSP Framework

Problem Statement Fake News Detection is to verify the authenticity of a given news article, we take it as a binary classification problem, where each sample is annotated with a ground truth label indicating its authenticity. Formally, Dataset $\mathcal{D}$ consists of N samples and each sample is represented by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{n, c_1, ..., c_N\}$ represents the news n and its comments $(c_1, ..., c_N)$, $\mathcal{E}$ represents a set of explicit interactive behaviors (e.g., retweet). The task objective of fake news detection is to learn a classifier f to classify samples and determine whether the news is true (labeled as 0) or false (labeled as 1), i.e.,

$$ f : \mathcal{G} \longrightarrow y, \quad y \in \{0, 1\}. \quad (1) $$

3.1 Overview

As illustrated in Figure 2, StruSP consists of two key components: bidirectional evolutionary propagation (BEP) learning and structure-aware propagation enhancement (SPE). The BEP module includes: (1) a structure-aware hybrid sampling strategy that traverses the propagation graph to capture both breadth-wise expansion and depth-wise progression patterns; and (2) a masked propagation modeling objective, which trains the LLM to reconstruct masked nodes along sampled propagation paths, thereby capturing structural dependencies within the diffusion process. In the SPE module, the trained LLM is used to generate structurally coherent extensions based on incomplete propagation trees. The synthetic propagation is then integrated into the original propagation structure for enhanced fake news detection.

3.2 Bidirectional Evolutionary Propagation Learning

To better model the structural dynamics of news propagation and support realistic propagation generation, we propose a bidirectional evolution propagation learning strategy (BEP), which comprises a structure-aware hybrid sampling mechanism and a direction-aware masked node prediction objective.

3.2.1 Structure-aware Hybrid Sampling

In real-world scenarios, news propagation typically exhibits complex bidirectional dynamics. On one hand, breadth-wise expansion emerges through wide dissemination across social networks (e.g., being shared among diverse user communities), resulting in multi-branched propagation structures. On the other hand, depth-wise progression arises from layered discussions (e.g., multiple rounds of comments and interactions), forming deep sequential chains. Inspired by Tan et al. (2023), we adopt a structure-aware hybrid sampling to model propagation dynamics along both dimensions. Specifically, given a propagation graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we utilize two graph traversal strategies: Breadth-First Search (BFS) and Depth-First Search (DFS) to encode the bidirectional evolution of the propagation tree:

$$P_{\mathcal{G}}^B = BFS(\mathcal{G}), \quad P_{\mathcal{G}}^D = DFS(\mathcal{G}), \quad (2)$$

where $P_{\mathcal{G}}^B$ captures breadth-wise expansion of $\mathcal{G}$, while $P_{\mathcal{G}}^D$ models depth-wise progression of $\mathcal{G}$. By encoding the propagation tree of each news, we obtain two sets of propagation paths: breadth-wise expansion paths ($\mathcal{P}_B = \{P_{\mathcal{G}}^B, \mathcal{G} \in \mathcal{D}\}$) and depth-wise progression paths ($\mathcal{P}_D = \{P_{\mathcal{G}}^D, \mathcal{G} \in \mathcal{D}\}$). Each propagation path $P \in \mathcal{P} = \mathcal{P}_D \cup \mathcal{P}_B$ is represented as a sequence of traversal nodes:

$$\begin{aligned} P &= (v_0, v_1, ..., v_{|P|}), \\ v_i &= <id_{parent}, i, c_i>, \end{aligned} \quad (3)$$

where each traversal node v_i is represented as a triple [1], $id_{parent} \in \{0, 1, ..., i-1\}$ represents the parent node index of v_i and c_i indicates the content of v_i.

3.2.2 Masked Propagation Modeling

While LLMs demonstrate remarkable generalization abilities, domain-specific fine-tuning remains crucial to adapt their broad linguistic knowledge to the unique structural patterns of propagation graphs.

---

[1] $v_0 = (\text{None}, 0, [\text{news content}])$ denotes the news node in the path.

Drawing inspiration from masked language modeling approaches, we implement a context modeling objective called masked propagation modeling that enhances the LLM's ability to capture the hierarchical evolution patterns inherent in real-world propagation. Specifically, we iteratively select node $v_m(m \in \{1, 2, ..., |P|\}$ from $P \in \mathcal{P}$ and mask its preceding and subsequent nodes respectively, generating two sub-paths P_m^{-1} (predict the preceding node v_{m-1}) and $P^{+1}m$ (predict the subsequent node v_{m+1}) from P:

$$\begin{aligned} P_m^{-1} &= (v_0, ..., v_{m-2}, v_{[\text{mask}]}, v_m), \\ P_m^{+1} &= (v_0, ..., v_{m-1}, v_m, v_{[\text{mask}]}), \end{aligned} \quad (4)$$

where $v_{[\text{mask}]}$ represents the node to be predicted. These sub-paths $P_m^z(z \in \{-1, +1\}$ are then textualized using a predefined prompt template $\Phi_1(\cdot)$ to create training samples:

> Given the propagation tree: P_m^z, please predict the masked comment node ({'parent node index': '[masked]', 'node index': '[masked]', 'content': '[masked]'}) in a JSON format as same as other nodes, i.e.,{parent node index: num, node index: num, content: text}.

The objective essentially constitutes a next-token prediction problem, where the LLM predicts the masked node $v_{[\text{mask}]}$ given input $\Phi_1(P_m^z)$ and compares it with the ground truth node v_{m+z}. The optimization objective is formulated as:

$$\begin{aligned} \mathcal{L} = \sum_{P \in \mathcal{P}} \Big(\sum_{m=0}^{|P|-1} &- \log \mathbb{P}(v_{m+1}|\Phi_1(P_m^{+1}))) \\ - \sum_{m=2}^{|P|} &\log \mathbb{P}(v_{m-1}|\Phi_1(P_m^{-1}))). \end{aligned} \quad (5)$$

3.3 Structure-aware Propagation Enhancement

Unlike prior approaches (Wan et al., 2024; Nan et al., 2024) that simulate propagation solely based on news content, we generate propagation under the guidance of existing propagation using the BEP-trained LLM. This allows structurally consistent extensions that better reflect real-world propagation patterns and improve downstream detection effectiveness.

Starting with a given news propagation $\mathcal{G}_0'$, we traverse it into a sequence $P_{\mathcal{G}',0}$ like the ones in Equation 3 by time order. And then the trained

Datasets	Twitter	CED	PHEME5
Number of News	1,154	3,387	5,801
Number of True News	575	1,538	3,829
Number of False News	579	1,849	1,973
Number of Comments	59,255	1,275,180	85,408
Average number of Comments	52	377	15

Table 1: The statistics of datasets.

LLM iteratively generates a comment node, which is added to the propagation sequence:

$$v_i' = \text{LLM}(\Phi_2(P_{\mathcal{G}',i-1})),$$
$$P_{\mathcal{G}',i} = P_{\mathcal{G}',i-1} \cup \{v_i'\}, \tag{6}$$

where LLM refers to the BEP-trained LLM, v_i' represents the generated traversal node as shown in Equation 3. $\Phi_2(\cdot)$ is the function that encodes $P_{n,v_{i-1}}'$ into a textual sequence following a predefined prompt template :

> Given the propagation tree: $P_{\mathcal{G}',i-1}$, please predict the next comment node in a JSON format as same as other nodes, i.e.,{parent node index: num, node index: num, content: text}.

3.4 Propagation-enhanced Detection

Ultimately, we reconstruct the enriched propagation tree $\mathcal{G}_k'$ by aggregating all node information in $P_{\mathcal{G}',k}$ from Equation 6, where k specifies the predefined scale of the number of nodes to generate.

In downstream fake news detection, we use the trained detector $f(\cdot)$ to detect and predict the authenticity label $\hat{y}$ of news using the enriched propagation $\mathcal{G}_k'$:

$$\hat{y} = f(\mathcal{G}_k'). \tag{7}$$

4 Experiments Setups

4.1 Datasets

We conduct experiments on three public datasets: **Twitter** (Siska et al., 2024), **CED** (Song et al., 2019) and **PHEME5** (Zubiaga et al., 2016). **Twitter** contains tweets published on Twitter[2], and each tweet is annotated with true of false. **CED** contains Chinese rumor data scraped from Weibo, including forwarding and comment information related to the original Weibo posts. **PHEME5** contains collections of rumors and non-rumors released on Twitter during 5 emergency events between 2014 and 2016. The statistics of the three datasets are

shown in Table 1. Following Chen et al. (2025), we divided the datasets into training, validation, and testing sets in a ratio of 7:1:2.

4.2 Evaluation Metrics

We evaluate our approach using two categories of metrics. For detection performance, we employ standard classification metrics including Accuracy, Macro-F1, Precision, Recall, and Area Under the ROC Curve (AUC).

To assess the quality of synthetic propagation, we utilize both structural and semantic metrics. The structural metrics comprise Structural Entropy (SE), Maximum Depth (MD), and Maximum Breadth (MB), which capture the topological characteristics of propagation trees. The semantic metrics include Semantic Consistency (SemC), Sentiment Consistency (SenC), and Semantic Homogeneity (SemH), which measure the coherence of textual content within the propagation. Detailed definitions of these propagation evaluation metrics are provided in Appendix A.

4.3 Baseline

For the evaluation of our propagation generation methods, we employ the following approaches:

BERT (Devlin et al., 2019) is a widely used pretrained language model for fake news detection, with the output from the last layer commonly fed into a classifier. **dEFEND** (Shu et al., 2019) develops a sentence-comment co-attention sub-network for fake news detection. **GCN** (Kipf and Welling, 2016) applies graph convolutional operations on the news propagation graph to learn news representations. **Bi-GCN** (Bian et al., 2020) models bidirectional propagation graphs based on the news propagation graph for detection. **EBGCN** (Wei et al., 2021) learns structural features from uncertain propagation using Bayesian graph convolutional networks. **RAGCL** (Cui and Jia, 2024) learns robust rumor representations through adaptive propagation graph contrastive learning. We utilize the above four propagation-based detection models to evaluate the effectiveness of synthetic propagation for detection. **GenFEND** (Nan et al., 2024) obtains 30 specific user profiles from three perspectives: gender, age, and education level. Then, LLMs are made to act as these thirty users to comment on news articles. **DELL** (Wan et al., 2024) makes LLMs act as designated users to comment on news articles or reply to other comments through an iterative process, thereby generating

---

[2]In July 2023, Twitter has been rebranded to X.

Methods	Twitter					PHEME5				
	Accuracy	Macro-F1	Recall	Precision	AUC	Accuracy	Macro-F1	Recall	Precision	AUC
BERT	71.12	70.86	71.44	71.83	71.13	81.83	79.22	85.56	87.63	78.88
LLM$_{text}$	56.00	54.60	53.20	55.23	57.24	33.74	28.64	32.78	34.14	31.78
dEFEND	75.12	73.56	75.56	75.89	75.13	83.24	82.45	83.54	92.45	89.42
LLM$_{comments}$	60.75	60.47	59.47	65.23	61.27	53.51	52.77	54.56	60.27	54.18
GCN	78.02	77.67	80.59	75.39	86.61	80.29	75.16	81.85	91.07	86.96
BiGCN	82.76	82.71	84.51	81.09	90.67	82.70	80.28	86.44	87.86	88.44
EBGCN	83.19	82.60	82.13	83.44	91.32	83.89	80.72	85.51	90.71	89.14
RAGCL	84.05	83.72	83.76	85.96	90.88	84.81	82.11	87.29	89.82	89.25
LLM$_{propagation}$	48.84	46.76	40.24	49.47	47.72	52.14	52.04	53.94	54.57	55.62
GenFEND										
w/BERT	78.26	75.64	78.12	76.64	74.78	83.84	81.23	87.57	89.64	87.89
w/dEFEND	82.25	82.04	83.54	81.42	90.70	85.07	84.66	86.75	92.46	91.63
DELL										
w/*single*	80.17	79.75	81.61	78.91	88.39	82.60	80.75	88.84	84.54	88.89
w/*vanilla*	80.17	82.55	83.34	76.14	90.75	82.52	80.60	88.50	84.80	89.20
w/*confidence*	78.97	78.10	83.14	77.34	90.21	83.03	81.08	88.62	85.60	89.07
w/*selective*	81.22	81.02	81.06	78.96	86.79	83.29	82.75	88.80	89.54	89.02
StruSP (Ours)										
w/GCN	81.03	79.75	81.61	78.91	88.39	81.54	80.24	86.36	88.41	80.65
w/BiGCN	84.04	83.78	**85.68**	84.59	92.08	84.45	83.25	87.85	88.96	90.70
w/EBGCN	84.48	84.23	84.60	84.85	**92.56**	86.21	84.96	89.88	93.12	90.38
w/RAGCL	**85.43**	**84.84**	85.05	**86.22**	92.45	**87.76**	**85.43**	**90.58**	**93.17**	**92.71**

Table 2: Results (%) of general fake news detection on Twitter and PHEME5. For each method, we run it five times and report the average results. The results of methods enhanced by StruSP are statistically significant than its baseline model (p-value < 0.05). The best results on each metric are in **boldface**.

propagation.

Additionally, following Chen et al. (2025), we evaluate LLMs as fake news detectors, categorizing the models into three types based on input content: LLMtext, LLMcomments, and LLM$_{propagation}$.

4.4 Implementation Details

All experiments are conducted on a single NVIDIA A40 GPU with 46GB of memory. The predefined number of generated nodes is set to 30. We implement all baseline methods under the same environment, following the parameter configurations reported in their original papers. Two large language models are used in our study: LLaMa3-8B-Instruct and Qwen3-4B. Unless otherwise specified, we report the results of LLM-based methods using LLaMa3-8B-Instruct.

For training the LLM backbone in StruSP, we construct a joint training set by merging the training portions of the Twitter and PHEME5 datasets. We adopt a parameter-efficient fine-tuning approach using LoRA (Hu et al.) with a rank of 8, applied to all transformer layers. The model is optimized using the AdamW optimizer with a cosine learning rate schedule, a base learning rate of 5e-5, and a warmup ratio of 0.1. All LLM backbones are trained for 4 epochs with Brain Floating Point 16-bit (BF16) precision enabled.

5 Experimental Results

We evaluate the effectiveness of StruSP on three real-world fake news datasets across different detection scenarios (Section 5.1) and conduct ablation studies to evaluate the effectiveness of each component in StruSP (Section 5.2). We further analyze the structural differences between synthetic and real propagation at two levels (Section 5.3): a macro-level analysis, which compares average metric values across samples, and a micro-level analysis, which examines their distribution at the individual sample level. We replace the LLM backbone of StruSP to investigate the impact of LLM choice on the framework's performance (Section 5.4).

5.1 Main Results

5.1.1 General Detection

Table 2 shows the performance of baselines and our method in general detection. Our method effectively enhances existing fake news detection methods. Specifically, StruSP *w*/RAGCL achieves the state-of-the-art performance on both datasets, and it gains 2.95% improvement in accuracy compared to RAGCL on PHEME5.

From the results, we have the following observations. First, compared to GenFEDN and DELL, our propagation-enhanced method performs bet-
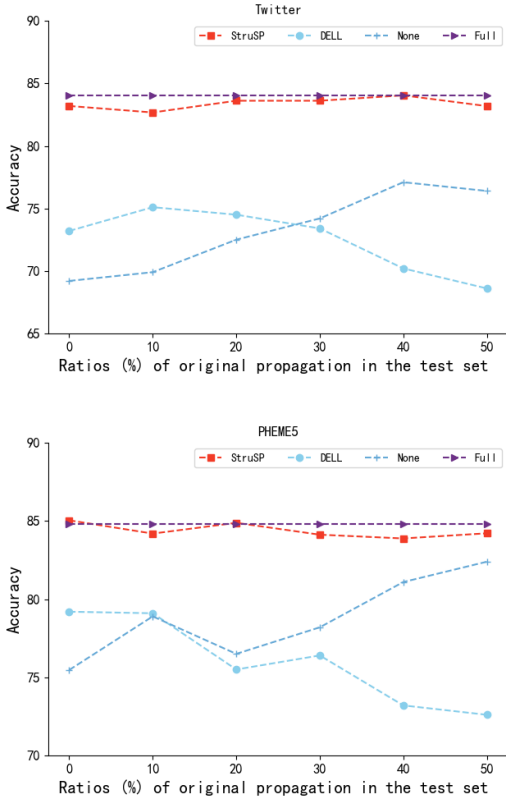
Figure 3: Results of StruSP and comparison methods on early detection. RAGCL is the backbone model. Full refers to using all available propagation data for prediction, i.e., general detection. None refers to only using limited real-world propagation for early detection. DELL and StruSP use both limited real-world and LLM-generated propagation for early detection.

ter, indicating that StruSP generates more informative and structurally realistic propagation trees that empower strong baselines again. Additionally, across all backbone models (GCN, BiGCN, EBGCN, RAGCL), integrating StruSP consistently boosts performance on both Twitter and PHEME5 datasets. This demonstrates that synthetic propagation generated by StruSP effectively complements real propagation in detection.

5.1.2 Early Detection

To evaluate the effectiveness of our method in early detection, where only limited propagation data is available, we test RAGCL (Cui and Jia, 2024) trained on full propagation data. During testing, only a fixed proportion of the propagation structure is retained to simulate early-stage scenarios (Chen et al., 2024). We utilize different methods to enrich the early-stage propagation for testing.

Figure 3 shows early detection results of our StruSP and comparison methods under the RAGCL

Methods	Twitter $\longrightarrow$ CED		PHEME5 $\longrightarrow$ CED	
	Accuracy	Macro-F1	Accuracy	Macro-F1
RAGCL	89.47	88.94	73.23	71.68
w/DELL	88.21	86.34	78.48	76.92
w/**StruSP**	**92.28**	**91.56**	**82.69**	**80.65**

Table 3: Performance of StruSP and other comparison methods in cross-domain detection. RAGCL is the backbone model. *Twitter $\longrightarrow$ CED* refers to training on the source domain (i.e., Twitter) and testing on the target domain (i.e., CED).

base model. The results indicate that: 1) StruSP effectively complements early-stage propagation, achieving detection performance close to the Full setting. This demonstrates its ability to generate structurally and semantically consistent propagation aligned with real-world dynamics. 2) DELL-generated propagation hinders detection performance, highlighting a mismatch between its role-playing generation and actual propagation patterns. This underscores the advantage of structure-aware generation in enhancing early detection.

5.1.3 Cross-platform Detection

To investigate generalization ability of StruSP-enhanced fake news detectors in the cross-platform detection, where there is little propagation data on some platform for the platform-specific detectors, we train RAGCL on two English Twitter datasets separately and test it on the CED dataset from Weibo platform, with the test samples translated into English[3]. We compare StruSP with DELL by using both to generate propagation data for CED and then perform detection on the generated propagation trees.

As shown in Table 3, RAGCL with StruSP achieves the best performance, demonstrating its ability to retain source-platform (Twitter) propagation dynamics while adapting to target content (Weibo). In contrast, using the original or DELL-generated propagation leads to poor performance, highlighting the difficulty of direct transfer and the importance of structure-aware generation for cross-platform fake news detection.

5.2 Ablation Study

To validate the effectiveness of StruSP, we conduct an ablation study of four ablative versions by removing structure-aware hybrid sampling(*w*/o SHS), masked propagation modeling (*w*/o MPM),

---

[3]To avoid differences caused by different languages, we translate the text of the samples in the CED test set into English with LLaMa3-8B-Instruct .

Methods	Twitter		PHEME5	
	Accuracy	Macro-F1	Accuracy	Macro-F1
StruSP	**85.43**	**84.84**	**87.76**	**85.43**
w/o SHS	84.48	84.25	85.78	85.14
w/o MPM	84.56	84.12	85.60	84.02
w/o BEP	83.62	83.22	83.55	81.06
w/o SPE	84.48	84.24	86.12	85.41

Table 4: Results (%) comparison between StruSP and its ablative variants. RAGCL is the backbone model.

Methods	Structural Metrics			Semantic Metrics		
	SE	MD	MB	SemC ↑	SenC↑	SemH
Orginal	0.94	3.56	14.17	-	-	0.85
GenFEND	-	-	-	0.89	0.53	0.91
DELL	1.75	4.71	10.09	0.91	0.50	0.92
LLM	1.54	4.10	11.13	0.94	0.79	0.88
StruSP	0.84	3.99	13.72	0.97	0.85	0.86

Table 5: Results of macro-level propagation analysis on the combined Twitter and PHEME5 datasets. Structural Entropy (SE), Maximum Depth (MD), and Maximum Breadth (MB) evaluate the structural features of propagation. Semantic Consistency (SemC), Sentiment Consistency (SenC), and Semantic Homogeneity (SemH) evaluate the semantic features of propagation.

bidirectional evolutionary propagation learning strategy (*w*/o BEP), and structure-aware propagation enhancement (*w*/o SPE). The results are shown in Table 4. It can be observed that the full StruSP *w*/RAGCL achieves better performance on both datasets. The performance drop of *w*/o SHS on both datasets demonstrates the importance of modeling bidirectional evolutionary propagation. Similarly, *w*/o MPM shows degraded results, indicating that providing LLMs with contextual propagation paths aids in capturing propagation dynamics. And the removal of both modules (*w*/o BEP) leads to the most significant decline, confirming that the two components are complementary and jointly crucial for realistic propagation generation. Furthermore, removing the structure-aware propagation enhancement module (*w*/o SPE) results in a noticeable performance drop, indicating that generating propagation solely from news content is less effective. This confirms that leveraging partial real propagation as guidance leads to more informative and structurally aligned synthetic propagation for detection.

5.3 Propagation Evaluation

We evaluate the quality of propagation generated by StruSP and other comparison methods. **LLM** refers to the use of an unfine-tuned LLaMa3-8B-Instruct model to generate propagation, following

(a) Structure Entropy   (b) Max Depth

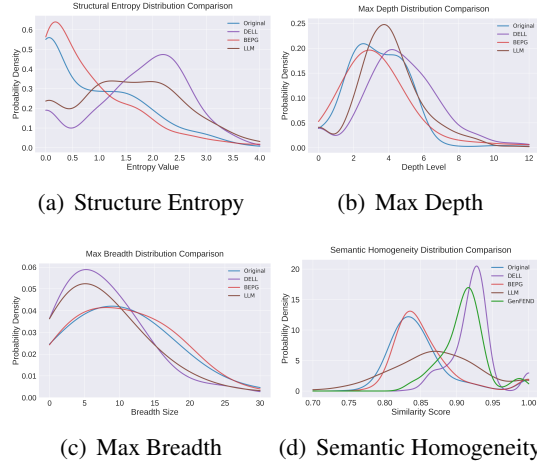(c) Max Breadth   (d) Semantic Homogeneity

Figure 4: Results of micro-level propagation analysis on the combined Twitter and PHEME5 datasets. The distributions of Structural Entropy, Instance-level Depth/Breadth, and Semantic Homogeneity across propagation generated by different methods

the approach described in Section 3.3. We conduct both macro-level and micro-level analyses. The macro-level evaluation assesses the overall structural and semantic similarity to real-world propagation, while the micro-level evaluation focuses on intra-propagation variation. Detailed definitions of all evaluation metrics are provided in Appendix A.

The evaluation results are shown in Table 5 and Figure 4. It can be observed that synthetic propagation generated by StruSP best aligns with real-world propagation, outperforming non-fine-tuned LLMs and role-playing methods (e.g., GenFEND, DELL) in both structural and semantic metrics. It indicates that the effectiveness of StruSP in guiding LLMs to generate realistic and informative propagation. Moreover, StruSP shows higher similarity to real propagation across semantic metrics compared to non-fine-tuned baselines, confirming the benefit of incorporating real-world propagation signals during training.

5.4 Performance with Different LLM Backbones

To validate the generalization of our method for different LLMs, we compare the performance of StruSP with LLaMa3-8B-Instruct and Qwen3-4B as the backbone and conduct a comprehensive performance comparison. As shown in Table 6, LLaMa3-8B-Instruct and Qwen3-4B achieved comparable performance. It indicates that StruSP is robust to the choice of backbone LLM, as both LLaMa3-8B-Instruct and Qwen3-4B can effec-

Method	Twitter		PHEME5	
	Accuracy	Macro-F1	Accuracy	Macro-F1
LLaMa3-8B as backbone				
GCN	81.03	79.75	81.54	80.24
BiGCN	84.04	83.78	84.45	83.25
EBGCN	84.48	84.23	86.21	84.96
RAGCL	**85.43**	**84.84**	**87.76**	**85.43**
Qwen3-4B as backbone				
GCN	80.72	78.14	81.82	80.56
BiGCN	83.62	83.19	84.07	83.16
EBGCN	84.48	84.23	85.96	84.87
RAGCL	**84.91**	**84.42**	**87.07**	**86.48**

Table 6: Results (%) of StruSP with different LLM backbones and GNN variants in general fake news detection. All models are evaluated on Twitter and PHEME5 datasets.

tively generate synthetic propagation data to enhance fake news detection. Moreover, our method shows the best detection performance with the different LLM backbones.

6 Conclusion

This paper proposes a structure-aware synthetic propagation enhanced fake news detection framework (StruSP). By employing a bidirectional evolutionary propagation learning strategy, StruSP enables LLMs to generate realistic and informative propagation trees and enrich the existing incomplete propagation tree. Experiments on three datasets demonstrate that StruSP significantly improves fake news detection performance in different incomplete propagation settings and produces realistic and diverse propagation.

7 Acknowledgements

Limitations

As an initial attempt to generate synthetic propagation that structurally aligns with real-world diffusion patterns, the proposed StruSP framework presents several limitations. First, it relies on observed propagation data to train the LLM-based generator, which makes its performance sensitive to the quality and coverage of the training data. Second, the generation quality is closely tied to

prompt design; less expressive or overly generic prompts may struggle to capture complex structural dependencies. Lastly, StruSP does not explicitly model user-level behaviors or social dynamics, potentially limiting the realism and personalization of the generated content, particularly in emotionally charged or user-driven conversations.

Ethics Statement

Our proposed method, StruSP, leverages large language models (LLMs) to generate synthetic propagation structures for the purpose of enhancing fake news detection under incomplete propagation scenarios. All generated content is used solely for research purposes and is not intended for public dissemination.

While our approach improves detection performance, we acknowledge the potential misuse of synthetic propagation generation for malicious purposes, such as falsifying social media diffusion. To mitigate this, we emphasize that our system is designed for controlled research and evaluation within the context of fake news detection.

We also recognize that LLM-generated content may reflect unintended biases or sentiments. To reduce this risk, we employ structure-aware training grounded in real-world data and evaluate outputs using semantic and sentiment alignment metrics.

Overall, this study aims to advance the understanding and mitigation of fake news on social platforms, and we encourage the responsible use of the techniques proposed.

References

Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. 2023. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351.

Tian Bian, Xi Xiao, Tingyang Xu, et al. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *AAAI*, 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.

Mengyang Chen, Lingwei Wei, Han Cao, Wei Zhou, and Songlin Hu. 2025. Explore the potential of llms in misinformation detection: An empirical study. In *AAAI 2025 Workshop on Preventing and Detecting LLM Misinformation (PDLM)*.

Mengyang Chen, Lingwei Wei, Han Cao, Wei Zhou, Zhou Yan, and Songlin Hu. 2024. Propagation structure-semantic transfer learning for robust fake news detection. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 227–244. Springer.

Minje Choi, Jiaxin Pei, Sagar Kumar, Chang Shu, and David Jurgens. 2023. Do llms understand social knowledge? evaluating the sociability of large language models with socket benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11370–11403.

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.

Carson Denison, Monte MacDiarmid, Fazl Barez, David Duvenaud, Shauna Kravec, Samuel Marks, Nicholas Schiefer, Ryan Soklaski, Alex Tamkin, Jared Kaplan, et al. 2024. Sycophancy to subterfuge: Investigating reward-tampering in large language models. *arXiv preprint arXiv:2406.10162*.

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*.

Yingtong Dou, Kai Shu, Congying Xia, Philip S. Yu, and Lichao Sun. 2021. User preference-aware fake news detection. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 2051–2055. ACM.

Robert Faris, Hal Roberts, Bruce Etling, Nikki Bourassa, Ethan Zuckerman, and Yochai Benkler. 2017. Partisanship, propaganda, and disinformation: Online media and the 2016 us presidential election. *Berkman Klein Center Research Publication*, 6.

Marc Fisher, John Woodrow Cox, and Peter Hermann. 2016. Pizzagate: From rumor, to hashtag, to gunfire in dc. *Washington Post*, 6:8410–8415.

Ivar Frisch and Mario Giulianelli. 2024. Llm agents in interaction: Measuring personality consistency and linguistic alignment in interacting populations of large language models. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*, pages 102–111.

Chen Gao, Xiaochong Lan, Zhi jie Lu, Jinzhu Mao, Jing Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. S3: Social-network simulation system with large language model-empowered agents. *ArXiv*, abs/2307.14984.

Suhaib Kh Hamed, Mohd Juzaiddin Ab Aziz, and Mohd Ridzwan Yaakub. 2023. Fake news detection model on social media by leveraging sentiment analysis of news content and emotion analysis of users' comments. *Sensors*, 23(4):1748.

D. Hu, L. Wei, W. Zhou, X. Huai, J Han, and S. Hu. 2021. A rumor detection approach based on multi-relational propagation tree. *Journal of Computer Research and Development*, 58(7):1395–1411.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Julie Jiang and Emilio Ferrara. 2023. Social-llm: Modeling user behavior at scale using language models and social network data. *ArXiv*, abs/2401.00893.

Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. 2019. exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, 9(19):4062.

Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. 2021. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia tools and applications*, 80(8):11765–11788.

Hamid Karimi and Jiliang Tang. 2019. Learning hierarchical discourse-level structure for fake news detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3432–3442.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

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*.

Yuhan Liu, Xiuying Chen, Xiaoqing Zhang, Xing Gao, Ji Zhang, and Rui Yan. 2024. From skepticism to acceptance: Simulating the attitude dynamics toward fake news. In *International Joint Conference on Artificial Intelligence*.

Yi-Ju Lu and Cheng te Li. 2020. Gcan: Graph-aware co-attention networks for explainable fake news detection on social media. In *Annual Meeting of the Association for Computational Linguistics*.

Alex Munyole Luvembe, Weimin Li, Shaohua Li, et al. 2023. Dual emotion based fake news detection: A deep attention-weight update approach. *IPM*, 60(4):103354.

Guanghui Ma, Chunming Hu, Ling Ge, et al. 2022. Towards robust false information detection on social networks with contrastive learning. In *CIKM*, pages 1441–1450.

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*.

Jing Ma, Wei Gao, Zhongyu Wei, et al. 2015. Detect rumors using time series of social context information on microblogging websites. In *CIKM*, pages 1751–1754.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1980–1989. Association for Computational Linguistics.

Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez, and David Vilares. 2023. Contrasting linguistic patterns in human and llm-generated news text. *Artificial Intelligence Review*, 57.

Qiong Nan, Qiang Sheng, Juan Cao, Beizhe Hu, Danding Wang, and Jintao Li. 2024. Let silence speak: Enhancing fake news detection with generated comments from large language models. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 1732–1742.

Kashyap Popat. 2017. Assessing the credibility of claims on the web. In *WWW*, pages 735–739.

Zhongyi Qiu, Hanjia Lyu, Wei Xiong, and Jiebo Luo. 2025. Can llms simulate social media engagement? a study on action-guided response generation. *arXiv preprint arXiv:2502.12073*.

Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. Csi: A hybrid deep model for fake news detection. In *CIKM*, pages 797–806.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Esin DURMUS, Zac Hatfield-Dodds, Scott R Johnston, Shauna M Kravec, et al. Towards understanding sycophancy in language models. In *The Twelfth International Conference on Learning Representations*.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 395–405.

Charlotte Siska, Katerina Marazopoulou, Melissa Ailem, and James Bono. 2024. Examining the robustness of LLM evaluation to the distributional assumptions of benchmarks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10406–10421, Bangkok, Thailand. Association for Computational Linguistics.

Changhe Song, Cheng Yang, Huimin Chen, et al. 2019. Ced: Credible early detection of social media rumors. *TKDE*, 33(8):3035–3047.

Xing Su, Jian Yang, Jia Wu, and Yuchen Zhang. 2023. Mining user-aware multi-relations for fake news detection in large scale online social networks. In *Proceedings of the sixteenth ACM international conference on web search and data mining*, pages 51–59.

Yanchao Tan, Zihao Zhou, Hang Lv, Weiming Liu, and Carl Yang. 2023. Walklm: A uniform language model fine-tuning framework for attributed graph embedding. *Advances in neural information processing systems*, 36:13308–13325.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *science*, 359(6380):1146–1151.

Herun Wan, Shangbin Feng, Zhaoxuan Tan, Heng Wang, Yulia Tsvetkov, and Minnan Luo. 2024. DELL: Generating reactions and explanations for LLM-based misinformation detection. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2637–2667, Bangkok, Thailand. Association for Computational Linguistics.

Lingwei Wei, Dou Hu, Wei Zhou, and Songlin Hu. 2024. Transferring structure knowledge: A new task to fake news detection towards cold-start propagation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8045–8049. IEEE.

Lingwei Wei, Dou Hu, Wei Zhou, et al. 2021. Towards propagation uncertainty: Edge-enhanced Bayesian graph convolutional networks for rumor detection. In *ACL*, pages 3845–3854.

Lingwei Wei, Dou Hu, Wei Zhou, et al. 2022. Uncertainty-aware propagation structure reconstruction for fake news detection. In *COLING*, pages 2759–2768.

Jiaying Wu and Bryan Hooi. 2023. Decor: Degree-corrected social graph refinement for fake news detection. In *KDD*, pages 2582–2593.

Feng Yu, Qiang Liu, Shu Wu, et al. 2017. A convolutional approach for misinformation identification. In *IJCAI*, pages 3901–3907.

Chunyuan Yuan, Qianwen Ma, Wei Zhou, et al. 2020. Early detection of fake news by utilizing the credibility of news, publishers, and users based on weakly supervised learning. In *COLING*, pages 5444–5454.

Zhenrui Yue, Huimin Zeng, Yimeng Lu, Lanyu Shang, Yang Zhang, and Dong Wang. 2024. Evidence-driven retrieval augmented response generation for online

misinformation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5628–5643.

Xueyao Zhang, Juan Cao, Xirong Li, Qiang Sheng, Lei Zhong, and Kai Shu. 2021. Mining dual emotion for fake news detection. In *Proceedings of the web conference 2021*, pages 3465–3476.

Zilong Zhao, Jichang Zhao, Yukie Sano, Orr Levy, Hideki Takayasu, Misako Takayasu, Daqing Li, Junjie Wu, and Shlomo Havlin. 2020. Fake news propagates differently from real news even at early stages of spreading. *EPJ data science*, 9(1):7.

Arkaitz Zubiaga, Maria Liakata, and Rob Procter. 2016. Learning reporting dynamics during breaking news for rumour detection in social media.

A Propagation Evaluation Metrics

We provide detailed definitions and calculation methods for the metrics used in evaluations of propagation quality.

Structural Entropy (SE) quantifies uncertainty in node degree distribution using Shannon entropy:

$$SE = -\sum_k p_k \log p_k, \qquad (8)$$

where p_k is the proportion of nodes with degree k.

Max Depth (MD) measures the maximum depth of a single propagation instance.

Max Breadth (MB) measures the maximum number of nodes at any depth level.

Semantic Consistency (SemC) measures the average semantic alignment between the generated propagation tree $\mathcal{G}'$ and the corresponding original propagation tree $\mathcal{G}$ by comparing their aggregated semantic representations:

$$SemC = \frac{1}{N} \sum_{i=1}^{N} \cos\left(\frac{1}{|V^{(\mathcal{G}'_i)}|} \sum_{v \in V^{(\mathcal{G}'_i)}} emb(v), \right.$$
$$\left. \frac{1}{|V^{(\mathcal{G}_i)}|} \sum_{v \in V^{(\mathcal{G}_i)}} emb(v) \right),$$

where N denotes the number of samples. $V_i^{(\mathcal{G})}$ and $V_i^{(\mathcal{G}')}$ are the node sets of $\mathcal{G}_i$ and $\mathcal{G}'_i$ respectively. $emb(v)$ is the BERT-based embedding of node V. The cosine similarity is computed between the average embedding of each tree pair.

Sentiment Consistency (SenC) measures the alignment of overall sentiment across generated and original propagation. It reflects whether the generated propagation $\mathcal{G}'$ preserves the dominant sentiment polarity of the original ones $\mathcal{G}$.

$$SenC = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left(MajSent(\mathcal{G}'_i) = MajSent(\mathcal{G}_i) \right), \quad (9)$$

where N denotes the number of samples. $MajSent(\mathcal{G}_i)$ represents the majority sentiment label (e.g., Positive or Negative) of the i-th tree in the datasets. The function $\mathbb{I}(\cdot)$ is the indicator function that returns 1 if the two labels are equal, and 0 otherwise. Sentiment labels are obtained using a pretrained sentiment classifier[4] applied to each

---

[4]https://huggingface.co/distilbert/distilbert-base-uncased-finetuned-sst-2-english

comment in the propagation tree. The majority sentiment of a tree is determined by the most frequent label among its nodes.

Semantic Homogeneity (SemH) measures pairwise coherence among all comments in a propagation, the cosine similarity is used to calculate the semantic homogeneity:

$$SemH = \frac{2}{|V|(|V|-1)} \sum_{i<j} \cos\left(emb(v_i), emb(v_j) \right) \quad (10)$$

B Prompt Design and Robustness Analysis

B.1 Robustness Through Fine-tuning and Structure

Our framework achieves robustness against prompt variations through two core design principles:

1. Deep Structural Learning via BEP: The Bidirectional Evolutionary Propagation learning strategy (Section 3.2 in the main paper) explicitly trains the LLM to become an expert in propagation dynamics. This process deeply ingrains structural awareness into the model's parameters, making its behavior inherently stable and far less sensitive to minor prompt variations compared to methods relying solely on in-context learning.

2. Formal API Design: By treating prompts as a formal API with structured JSON I/O (detailed in Section B.2), we ensure that the LLM engages with the task's logic directly. This structured interface minimizes ambiguity and ensures consistent interpretation regardless of natural language variations in the prompt.

These design choices collectively ensure that our method's performance is robust and reliable across different prompt formulations.

B.2 Structured JSON I/O Format

Our framework employs a structured JSON format for both input and output:

> **Input:**
> Propagation tree with nodes: [{node_index: 0, parent_index: -1, content: "text"}, ...]
> **Output:**
> Predicted node: {parent_node_index: num, node_index: num, content: "text"}

This structured approach transforms the interaction into a clear, machine-readable function call, ensuring reliable and predictable outputs while minimizing ambiguity in the LLM's interpretation of the task.

Method	Accuracy ↑	Macro-F1 ↑	$|SE_{ori.} - SE_{syn.}|$↓	SemC ↑
StruSP w/P1 (Ours)	85.43	84.84	0.10	0.97
StruSP w/P2	84.91	84.45	0.14	0.97
StruSP w/P3	85.12	84.36	0.10	0.97
DELL	81.22	81.02	0.81	0.91
GenFEND	82.25	82.04	-	0.89

Table 7: Performance comparison across different prompt variants on the Twitter dataset. $|SE_{org.} - SE_{syn.}|$ denotes the absolute value of the difference between the structural entropy score of the original propagation ($SE_{ori.}$) and the structural entropy score of the synthetic propagation ($SE_{syn.}$). The results demonstrate that our BEP learning strategy ensures robust performance regardless of prompt formulation.

B.3 Prompt Sensitivity Analysis

To demonstrate the robustness of our Bidirectional Evolutionary Propagation (BEP) learning strategy against prompt variations, we conducted comprehensive experiments using three different prompt formulations on the Twitter dataset.

B.3.1 Prompt Variants

We evaluated the following prompt variants:

P1 (Structured - Ours): *"Given the propagation tree: {tree}, please predict the next comment node in a JSON format as same as other nodes, i.e., {parent node index: num, node index: num, content: text}."*

P2 (Minimal): *"Given the propagation tree: {tree}, please predict the next comment node."*

P3 (Detailed): *"Given the propagation tree: {tree}, please carefully analyze the structural patterns and semantic context, then predict the next comment node that maintains both structural consistency and semantic coherence in a JSON format, i.e., {parent node index: num, node index: num, content: text}."*

B.3.2 Experimental Results

Table 7 presents the performance comparison across different prompt variants, demonstrating the stability of our approach.

B.3.3 Analysis and Discussion

The experimental results reveal several key insights:

1. **Minimal Performance Variation:** The performance difference between prompt variants is marginal (less than 0.52% in ACC and 0.39% in Macro-F1), demonstrating that our BEP learning strategy successfully reduces sensitivity to prompt formulation.

2. **Consistent Structural Understanding:** All prompt variants maintain similar Structural Entropy and identical Semantic Consistency with original propagation, indicating that the model's understanding of propagation dynamics is deeply ingrained through fine-tuning rather than dependent on prompt engineering.

3. **Superiority over Baselines:** Even with the minimal prompt (P2), our method outperforms baseline approaches by significant margins, confirming that the robustness stems from our fine-tuning strategy rather than prompt sophistication.

C Computational Cost Analysis

We employed Parameter-Efficient Fine-Tuning (PEFT) using LoRA (rank=8) to significantly reduce the training cost while maintaining model performance. Fine-tuning the LLaMa3-8B-Instruct model on the combined Twitter and PHEME5 datasets took approximately 4 hours on a single NVIDIA A40 GPU, representing a one-time, manageable cost. This approach reduces the number of trainable parameters to approximately 0.1% of the original model size, making the fine-tuning process highly memory-efficient and accessible even in resource-constrained environments.

During inference, generating 30 synthetic comments for a single news propagation tree takes on average 1 minute on a single NVIDIA A40 GPU. This modest one-time training cost creates a powerful, reusable generator that can produce unlimited amounts of high-quality training data, enabling significant performance gains for lightweight GNN detectors in low-resource scenarios. For example, augmenting 1000 propagation trees would require approximately 16.7 GPU hours but would yield 30,000 high-quality training samples, demonstrating an excellent cost-benefit ratio for practical deployment.

D Generation Process and Automated Quality Assurance

Our generation process employs carefully selected parameters to balance diversity with coherence. We generate 30 nodes per propagation tree using the fine-tuned LLaMA3-8B model with LoRA adaptation. For decoding, we use nucleus sampling (top-p = 0.9) with a temperature of 0.6, allowing up to 3 retry attempts per node generation. These parameters were empirically determined to ensure syn-

thetic propagation patterns remain realistic while providing sufficient variability for effective data augmentation.

The quality of generated data is ensured through a rigorous automated validation pipeline, as detailed in Algorithm 1. Each generated node must pass through three sequential validation gates before being accepted into the propagation tree. The **Syntactic Filter** ensures the LLM's output is well-formed JSON, triggering re-generation for any malformed responses. The **Structural Filter** validates topological integrity by checking for valid parent references, preventing self-loops, ensuring unique node identifiers, and maintaining proper tree structure without cycles. Finally, the **Content Filter** ensures semantic quality by filtering out empty content, boilerplate refusal messages, repetitive text, and responses shorter than a minimum threshold.

Algorithm 1: Automated Node Validation Pipeline

---

1 Existing propagation tree $P_{current}$, $\text{LLM}_{generator}$, Max_retries $= 3$ A new valid node v_{new} or Failure GenerateValidNode$P_{current}$ **for** $i = 1$ *to Max_retries* $+ 1$ **do**

2 *// Generate a candidate node*
  raw_output $\leftarrow \text{LLM}_{generator}(P_{current})$

3 *// — Validation Gate 1: Syntactic Filter —*

4 **try:**
  node_json $\leftarrow$ ParseJSON(raw_output)
  except JSONDecodeError: **continue**
  Retry if output is not valid JSON

5 *// — Validation Gate 2: Structural Filter —*

6 **if** *not IsStructurallyValid(node_json, $P_{current}$)* **then**

7 **continue** Retry if parent_id or node_id is invalid

8 *// — Validation Gate 3: Content Filter —*

9 **if** *IsContentInvalid(node_json['content'])* **then**

10 **continue** Retry if content is empty, refusal, etc.

11 *// — Success: Node is valid —*

12 **return** CreateNode(node_json)

13 *// If all retries fail, return Failure*

14 **return** Failure