SINCon: Mitigate LLM-Generated Malicious Message Injection Attack for Rumor Detection

Mingqing Zhang[1,2*], Qiang Liu[1,2*], Xiang Tao[1,2], Shu Wu[1,2†], Liang Wang[1,2]
[1]New Laboratory of Pattern Recognition (NLPR)
State Key Laboratory of Multimodal Artificial Intelligence Systems
Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences
{mingqing.zhang, xiang.tao}@cripac.ia.ac.cn,
{qiang.liu, shu.wu, wangliang}@nlpr.ia.ac.cn

Abstract

In the era of rapidly evolving large language models (LLMs), state-of-the-art rumor detection systems, particularly those based on Message Propagation Trees (MPTs), which represent a conversation tree with the post as its root and the replies as its descendants, are facing increasing threats from adversarial attacks that leverage LLMs to generate and inject malicious messages. Existing methods are based on the assumption that different nodes exhibit varying degrees of influence on predictions. They define nodes with high predictive influence as important nodes and target them for attacks. If the model treats nodes' predictive influence more uniformly, attackers will find it harder to target high predictive influence nodes. In this paper, we propose **S**imilarizing the predictive **In**fluence of **N**odes with **Con**trastive Learning (**SINCon**), a defense mechanism that encourages the model to learn graph representations where nodes with varying importance have a more uniform influence on predictions. Extensive experiments on the Twitter and Weibo datasets demonstrate that **SINCon** not only preserves high classification accuracy on clean data but also significantly enhances resistance against LLM-driven message injection attacks.

1 Introduction

The rapid advancement of large language models (LLMs) has revolutionized natural language processing, enabling impressive capabilities in text generation (Li et al., 2024; Huang et al., 2024), summarization (Zhu et al., 2023; Xu et al., 2024), and contextual reasoning (Deng et al.; Kwon et al., 2024). However, these advancements also introduce new security challenges (Zhan et al., 2023), particularly in the domain of rumor detection on social media.

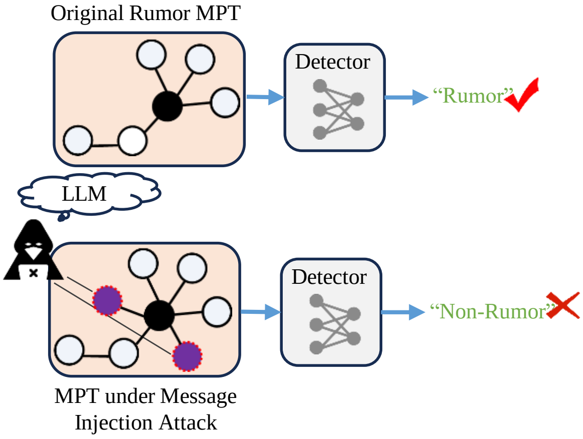

Figure 1: The rumor detection model is attacked by LLM-generated malicious message injection. The message injection attack, generated by an LLM, introduces new nodes and edges, altering the topology and semantics of the MPT. This causes the rumor detection model to fail in effectively detecting the rumor.

Recent studies have revealed that Message Propagation Trees (MPTs), modeled as conversation trees with the root representing the source post and subsequent nodes representing retweets or comments, are vulnerable to malicious message injection attacks when used in rumor detection models that leverage graph neural networks (GNNs) to analyze message propagation patterns (Liu and Wu, 2018; Zhang and Li, 2019; Song et al., 2021). Attackers can exploit LLMs to generate and inject deceptive messages into MPTs, significantly altering their topological and semantic structure. As a result, even state-of-the-art rumor detection models can be misled into classifying rumors as non-rumors, undermining their effectiveness in mitigating misinformation (Sun et al., 2024b; Li et al., 2025). As shown in Figure1, the attacker leverages LLM to conduct a message injection attack on MPTs, successfully bypassing the rumor detector.

Previous methods for attacking MPT-based rumor detection models rely on the assumption: **dif-**

[*] The first two authors contributed equally to this work.
[†]Corresponding Author

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12570–12581
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

ferent nodes in an MPT contribute unequally to the model's prediction, with important nodes having a greater predictive influence than unimportant ones (Mądry et al., 2017; Zou et al., 2021; Luo et al., 2024). Therefore, based on the node importance scores obtained through attribution approaches, the attack can be viewed as an iterative process where the most important nodes are targeted first. Consequently, the imbalance in predictive influences of nodes within the MPT leads to a critical vulnerability. Attackers can design targeted attacks that focus on high-influence nodes, triggering a chain reaction that disrupts the overall propagation structure.

Building on the aforementioned assumption, the success of attacks against MPT-based rumor detection models becomes clear. In a MPT, nodes can be categorized into important nodes, which carry more influential information for prediction, and unimportant nodes, which contribute less. Attack methods that target important nodes first are able to perturb the most critical information at each step, thereby making the model more susceptible to deception. Therefore, a counterintuitive question naturally arises: **Would the model be more robust if both important and unimportant nodes exerted a similar degree of influence on its predictions?**

To explore the aforementioned question, We propose **S**imilarizing the predictive **I**nfluence of **N**odes with **Con**trastive Learning (**SINCon**), a self-supervised regularization method designed to enhance model robustness against adversarial message injection attacks. SINCon mitigates the effect of localized perturbations by ensuring that both high- and low-influence nodes contribute more evenly to model predictions. Specifically, we define important and unimportant nodes as the top and bottom 10% of nodes ranked by influence scores within the MPT. To regularize the model, we introduce two data augmentation strategies: one that masks important nodes and another that masks unimportant nodes. SINCon then leverages a contrastive learning objective to (1) reduce the disparity in model predictions between these two augmented MPTs, ensuring that nodes of different influence levels have a more uniform impact, (2) maintain similarity between the augmented MPTs and the original MPT, preventing excessive information loss, (3) minimize the agreement between the original MPT and other distinct MPTs within the same batch, avoiding the trivial solution of pattern collapse and encouraging the model to learn more discriminative and robust representations.

We conduct extensive experiments on Twitter and Weibo datasets, evaluating SINCon against state-of-the-art MPT-based rumor detection models under LLM-generated malicious message injection attack. Our results demonstrate that by integrating SINCon into the training process, we effectively reduce the model's sensitivity to adversarial message injections, making it significantly more resilient to LLM-driven attacks while maintaining high performance on clean data.

Our main contributions can be summarized as follows:

- We identify the imbalance in node influence within MPT-based rumor detection models, which makes them vulnerable to malicious message injection attacks.
- We introduce SINCon, a contrastive learning method that balances node influence, reducing the model's vulnerability to attacks.
- Extensive experiments on Twitter and Weibo datasets show that SINCon improves model robustness to LLM-driven attacks while maintaining high performance on clean data.

2 Related Work

MPT-based Rumor Detection. Rumor detection on social media aims to identify and prevent the spread of misinformation. Recent methods use GNNs to capture information from MPTs (Wu et al., 2020; Xu et al., 2022; Zhang et al., 2023b; Wu et al., 2023; Tao et al., 2024a; Liu et al., 2024a). An MPT-based rumor detection model typically has three components: (1) message encoding, (2) GNN and (3) a readout function. Different studies use varied approaches for these components, such as word frequency counts (Malhotra and Vishwakarma, 2020; Khoo et al., 2020; Sun et al., 2022; Wu et al., 2023; Liu et al., 2024b) or dense embeddings for encoding (Liu et al., 2024c; Tao et al., 2024b; Zhang et al., 2024b; Sun et al., 2024a; Cui et al., 2022), and GCN or GAT for learning propagation patterns (Wu et al., 2022; Xu et al., 2022; Zhang et al., 2024a; Gong et al., 2024). The readout function often combines strategies like mean or max aggregation. In this paper, we mainly applied several state-of-the-art MPT-based rumor detection models to investigate their robustness.

LLM-Generated Attacks. Large Language Models (LLMs) are capable of generating highly coher-

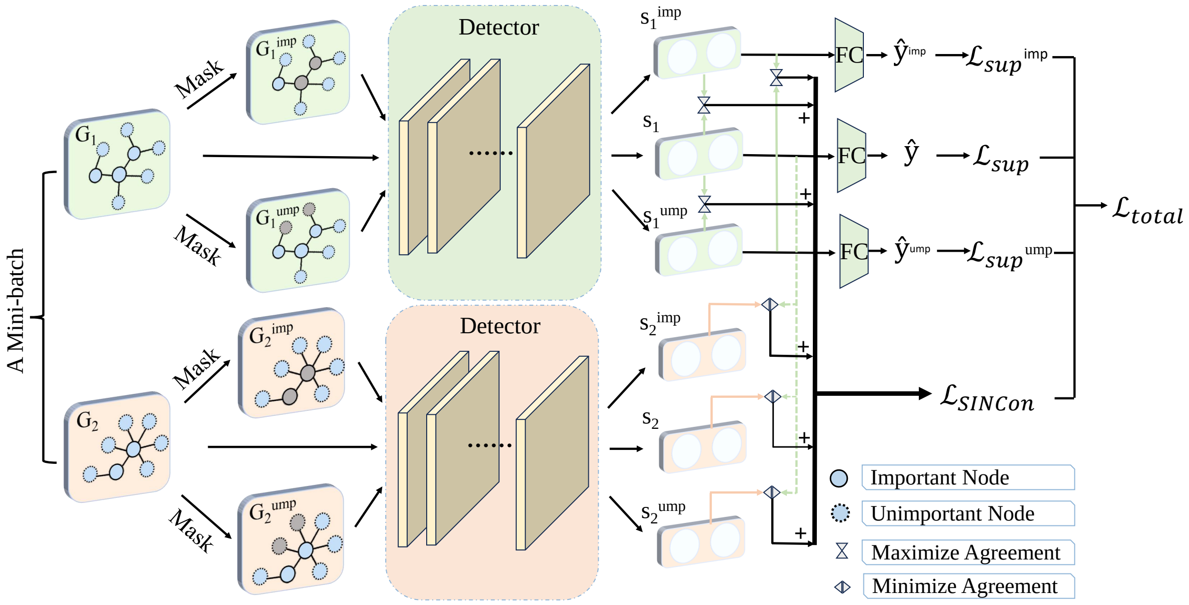

Figure 2: Architecture of SINCon. Given a mini-batch $G_i \in \{G_i\}_{i=1}^{B}$ of MPTs, where $B = 2$: (1) we define the top 10% of nodes with the highest and lowest influence scores in an MPT as important and unimportant nodes, respectively, based on Eq. 12. (2) To regularize the model, we introduce two data augmentation strategies: one that masks important nodes and another that masks unimportant nodes. (3) reduce the disparity in model predictions between these two augmented MPTs, maintain similarity between the augmented MPTs and the original MPT, minimize the agreement between the original MPT and other distinct MPTs within the same batch.

ent and contextually relevant text (Liu et al., 2023; Yang et al., 2024; Zhu et al., 2024; Valmeekam et al., 2023). However, while large language models demonstrate significant capabilities, they are increasingly drawing attention due to their generation of malicious information (Kreps et al., 2022). Recent research has shown that content generated by LLMs is often indistinguishable from content created by humans (Zhao et al., 2023; Uchendu et al., 2023).Some works considered the use of LLMs for rumor generation (Huang et al., 2022; Lucas et al., 2023; Pan et al., 2023). In this work, we aim to investigate methods for detecting such LLM-generated rumors and propose a defense mechanism to mitigate their impact on rumor detection systems.

Adversarial Attacks and Defenses in Rumor Detection. Rumor Detection Model adversarial attacks include evasion attacks (Luo et al., 2024) and poisoning (Li et al., 2023), as well as global (Fang et al., 2024) and targeted types (Zhang et al., 2023c). With the rapid development of LLM technology, LLMs have become tools for attackers (Hu et al., 2024; Xu et al., 2023). These attacks exploit the capabilities of LLMs to craft misleading messages or manipulate the structure of MPTs, resulting in subtle alterations to node features or edge relationships that deceive the model into making incorrect predictions. Adversarial samples are commonly used in various studies to train GNNs with enhanced robustness via adversarial training techniques (Gosch et al., 2024; Zhai et al., 2023; Zhang et al., 2023a). However, due to the scarcity of adversarial samples, the effectiveness of these methods is often limited. In this work, we introduce the technique of similarizing the influence of nodes with Contrastive Learning to enhance the robustness of rumor detection models.

3 Preliminaries

3.1 MPT-based Rumor Detection

Message Propagation Tree. Let $\mathcal{G} = \{G_i\}_{i=1}^{|\mathcal{G}|}$ be a set of MPTs, where each MPT $G_i = (\mathcal{X}_i, \mathbf{A}_i)$ consists of a set of messages $\mathcal{X}_i = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_{n_i}^{(i)}\}$, and an adjacency matrix $\mathbf{A}_i \in \{0,1\}^{n_i \times n_i}$ indicating reply or retweet relations. Here, $x_1^{(i)}$ is the source post and $\{x_j^{(i)}\}_{j=2}^{n_i}$, representing comments, replies, or retweets related to the source post, n_i denotes the number of messages in the i-th MPT. Each MPT has a binary label $y_i \in \{y_r, y_{nr}\}$, where y_r and y_{nr} represent rumor and non-rumor classifications, respectively.

We split the dataset into $\mathcal{G}_{\text{train}}$ and $\mathcal{G}_{\text{test}}$, corresponding to the training and testing sets of MPTs, respectively. The goal of MPT-based rumor detection is to train a binary classifier $f_\theta(G)$, parameterized by θ, using $\mathcal{G}_{\text{train}}$. The classifier $f_\theta(G)$ is trained on the training set $\mathcal{G}_{\text{train}}$ by minimizing the loss:

$$\mathcal{L}_{\text{sup}}(f_\theta(G)) = \sum_{G_i \in \mathcal{G}_{\text{train}}} \mathcal{L}(f_\theta(G_i), y_i), \quad (1)$$

with optimal parameters

$$\theta^* = \arg\min_\theta \mathcal{L}_{\text{sup}}(f_\theta(G)). \quad (2)$$

The trained model predicts $\hat{y}_i = f_{\theta^*}(G_i)$ for unseen MPTs in $\mathcal{G}_{\text{test}}$.

MPT-based Rumor Detector. Messages are encoded into feature vectors using an encoding function $\mathcal{E}(\cdot)$:

$$\mathbf{H}^{(0)} = \mathcal{E}(X) = [\mathbf{h}_1^{(0)}, \mathbf{h}_2^{(0)}, \ldots, \mathbf{h}_{n_i}^{(0)}]. \quad (3)$$

A GNN is then applied to learn both propagation patterns and content. For each message x_u, the feature update at layer l is:

$$\begin{aligned}\mathbf{h}_u^{(l)} =& \sigma^{(l-1)}\Big(\mathbf{h}_u^{(l-1)}, \\ & \mathcal{AGG}_{x_v \in N(x_u)}\left(\gamma^{(l-1)}(\mathbf{h}_v^{(l-1)}, \mathbf{h}_u^{(l-1)})\right)\Big),\end{aligned} \quad (4)$$

where $\sigma(\cdot)^{(l)}$ and $\gamma(\cdot)^{(l)}$ are the activation functions at the l-th layer of the GNN, $x_v \in N(x_u)$ is the 1-hop retweets or comments of message x_u, and $\mathcal{AGG}(\cdot)$ represents the aggregation operation. A readout function $\mathcal{R}(\cdot)$ aggregates these features into a summary representation:

$$\mathbf{s} = \mathcal{R}(\mathbf{H}^{(L)}). \quad (5)$$

Finally, the prediction is given by:

$$\hat{y} = \text{Softmax}(\mathbf{s}). \quad (6)$$

3.2 Message Injection Attack

Objective of the Attack. We denote $\mathcal{G}_r \subseteq \mathcal{G}_{\text{test}}$ and $\mathcal{G}_{nr} \subseteq \mathcal{G}_{\text{test}}$ as the set of rumor and non-rumor MPTs in the testing MPT set $\mathcal{G}_{\text{test}}$, respectively. The goal of the attack is to deceive the rumor detector into misclassifying a rumor MPT $G \in G_r$ as a non-rumor MPT by injecting a set of malicious messages $\mathcal{X}_{\text{atk}}$ into the MPT, with the constraint $|\mathcal{X}_{\text{atk}}| \leq \Delta$ and $d_{\text{in}}(x_u) = 1, \forall x_u \in \mathcal{X}_{\text{atk}}$. The budget Δ refers to the maximum number of malicious messages that can be injected into the MPT. This constraint ensures that the attack remains inconspicuous. The attacker minimizes the negative testing loss:

$$\min \sum_{G \in G_r} -\mathcal{L}_{\text{sup}}(f_\theta^*(G')), \quad (7)$$

where $G' = (\mathcal{X}', A')$ is the MPT with injected malicious messages.

Message Pair and Root-Centric Homophily. The attack effectiveness relies on disrupting the MPT homophily distribution. The message pair homophily between messages x_u and x_v is defined as:

$$\text{sim}(x_u, x_v) = \frac{\mathbf{h}_u^{(0)} \cdot \mathbf{h}_v^{(0)T}}{\|\mathbf{h}_u^{(0)}\|_2 \|\mathbf{h}_v^{(0)}\|_2}. \quad (8)$$

The root-centric homophily measures the similarity between the source post x_1 and other messages in the MPT:

$$\text{sim}_{\text{root}}(G) = \frac{1}{n} \sum_{x_j \in \{x_j\}_{j=2}^n} \text{sim}(x_j, x_1). \quad (9)$$

Iterative Malicious Message Generation. To generate malicious messages, we employ system prompt which is demonstrated in Appendix A.A.1. The process begins with the system prompt p and the source post x_1, producing an initial malicious message :

$$x_{\text{atk}} = \text{LLM}(x_1, p). \quad (10)$$

If the root-centric homophily of the generated message exceeds a threshold λ, the prompt is refined iteratively:

$$x'_{\text{atk}} = \text{LLM}(x_{\text{atk}}, p'), \quad (11)$$

where p' contains the homophily information, as detailed in Appendix A.2.

Connecting Malicious Messages. The more a message and its neighboring messages are commented on or retweeted, the higher the influence that message holds in the final summary representation $\mathbf{s}$ (Luo et al., 2024).The generated malicious messages are connected to existing messages in the MPT based on their influence score, which is calculated as:

$$I_{x_u} = \sqrt{d_{x_u} \cdot d_{x_v}}, \quad x_v \in \mathcal{N}(x_u) \cup \{x_u\}, \quad (12)$$

where I_{x_u} represents the influence score of node x_u, and d_{x_u} denotes the degree centrality of node x_u. The malicious message is then connected to the message with the highest influence score, updating the adjacency matrix A to A'.

Attack Procedure. For each rumor MPT $G \in \mathcal{G}_{\text{test}}$, the LLM-based Message Injection Attack generates malicious messages and injects them into the MPT. If the prediction of the MPT is a non-rumor $\hat{y} = \mathrm{y}_{nr}$, the message injection stops for that MPT.

4 Method

The architecture of SINCon is illustrated in Figure 2. Recall that the goal of SINCon is to similarize the influence of nodes. To formally define this goal, we first define the 10% of nodes in an MPT with the highest and lowest influence scores as the important and unimportant nodes, respectively, according to Eq.12. We then propose two data augmentation operations, $t^{imp}(\cdot)$ and $t^{ump}(\cdot)$, which respectively means mask important and unimportant nodes in a MPT. Therefore, under the training scenario of Eq.1, the primary goal of SINCon can now be formulated as:

$$\min_{\theta} \|\mathcal{Q}_{\text{imp}} - \mathcal{Q}_{\text{ump}}\| : \tag{13}$$

$$\mathcal{Q}_{\text{imp}} = \mathop{\mathbb{E}}_{G^{\text{imp}} \sim t^{\text{imp}}(G)} \Big[\mathcal{P}(G) - \mathcal{P}(G^{\text{imp}})\Big],$$
$$\mathcal{Q}_{\text{ump}} = \mathop{\mathbb{E}}_{G^{\text{ump}} \sim t^{\text{ump}}(G)} \Big[\mathcal{P}(G) - \mathcal{P}(G^{\text{ump}})\Big],$$

Here, G^{imp} is an augmentation sampled from $t^{\text{imp}}(G)$, and G^{ump} is an augmentation sampled from $t^{\text{ump}}(G)$. $\mathcal{P}(\cdot)$ represents the model's predicted probability distribution over the possible output classes for the input MPT. $\mathcal{Q}^{\text{imp}}$ and $\mathcal{Q}^{\text{ump}}$ measure the extent of model confidence decrease when information in the important and unimportant nodes is mask, indicating the overall influence of the information in nodes of different importance on prediction.

The complete objective of SINCon can be further decomposed into two perspectives:

Objective 1: The influence of different nodes should be similar, thus the model should treat the MPT with information in nodes of different importance mask (G^{imp} and G^{ump}) similarly.

Objective 2: The influence of different nodes should be slight, thus the model should treat the MPT with different information mask (G^{imp} and G^{ump}) similarly to the original MPT that contains complete information (G).

To achieve Objective 1 and Objective 2, and further the goal of SINCon, we use a contrastive loss objective from the perspective of MPT representation. To define the contrastive loss objective, for convenience, we first define the calculation S:

$$S^{(k,l)}_{(i,j)} = \exp\left(\mathrm{sim}[\mathbf{s}^k_i, \mathbf{s}^l_j]/\tau\right), \tag{14}$$

where $k, l \in \{\text{imp}, \text{ump}, \cdot\}$, respectively indicate the augmentation sampled from $t^{\text{imp}}(\cdot)$, the augmentation sampled from $t^{\text{ump}}(\cdot)$, and the normal example. i, j are the example indices, $\mathrm{sim}[\mathbf{r}_i, \mathbf{r}_j] = \mathbf{r}_i^\top \mathbf{r}_j / \|\mathbf{r}_i\| \|\mathbf{r}_j\|$ is the cosine similarity, and τ is a temperature parameter similar to the NT-Xent loss (Chen et al., 2020; Oord et al., 2018).

Then the contrastive loss function for an example in a mini-batch $G_i \in \{G_i\}_{i=1}^B$ is defined as:

$$\mathcal{L}_{\text{SINCon}}(G_i; \theta) = \mathop{\mathbb{E}}_{\substack{G_i^{\text{imp}} \sim t^{\text{imp}}(G_i) \\ G_i^{\text{ump}} \sim t^{\text{ump}}(G_i)}} \left[-\log \frac{S_{\text{positive}}}{\sum_{j=1}^{B} S_{\text{negative}}}\right], \tag{15}$$

where

$$\mathcal{S}_{\text{positive}} = \mathcal{S}^{(\text{imp},\text{ump})}_{(i,i)} + \mathcal{S}^{(\cdot,\text{ump})}_{(i,i)} + \mathcal{S}^{(\cdot,\text{imp})}_{(i,i)}, \tag{16}$$

$$\mathcal{S}_{\text{negative}} = \mathcal{S}^{(\cdot,\cdot)}_{(i,j)} + \mathbb{1}_{(i \neq j)} \cdot \left[\mathcal{S}^{(\cdot,\text{ump})}_{(i,j)} + \mathcal{S}^{(\cdot,\text{imp})}_{(i,j)}\right]. \tag{17}$$

Let B be the batch size, and $\mathbb{1}_{(\cdot)}$ be an indicator function that equals 1 if the condition $(\cdot)$ is true; otherwise, it equals 0. Specifically, to calculate the loss for each mini-batch, we first obtain the augmentations G_i^{ump} from $t^{\text{ump}}(G_i)$ and the augmentations G_i^{imp} from $t^{\text{imp}}(G_i)$ for each example in the mini-batch.

To achieve Objective 1, we use the term $S^{(\text{imp},\text{ump})}_{(i,i)}$ in the numerator. This constraint maximizes the similarity between the representations of the augmentations with important and unimportant nodes removed, making the different degrees of incomplete information in the augmentations have a similar impact on the prediction.

To achieve Objective 2, we use the terms $S^{(\cdot,\text{ump})}_{(i,i)}$ and $S^{(\cdot,\text{imp})}_{(i,i)}$ in the numerator. These constraints maximize the similarity between the original MPT and the two augmentations, ensuring that the incomplete information in the remaining nodes of the augmentations has a similar influence as the complete information in the normal MPT.

Intuitively, the semantics of different examples should be distinct. Following the constraints in S_{positive}, the semantics of the augmentations of different examples should also be different. Therefore, the three terms in S_{negative} indicate that, given an example within a mini-batch, both the other examples and the augmentations derived from other examples are treated as negative examples.

The final loss of SINCon regularization is computed across all examples in a mini-batch. When SINCon is used in the normal training scenario Eq.1, the overall objective is:

$$\begin{aligned} &\min_{\theta} \mathcal{L}_{\text{total}}(\theta) \\ &= \mathcal{L}_{\text{sup}}(f_{\theta}(G)) + \alpha_1(\mathcal{L}_{\text{sup}}(f_{\theta}(G^{imp})) \\ &+ \mathcal{L}_{\text{sup}}(f_{\theta}(G^{ump}))) + \alpha_2 \mathcal{L}_{\text{SINCon}}(G), \end{aligned} \tag{18}$$

where α_1 and α_2 are the parameters balancing the supervised part and the contrastive regularization part.

5 Experiment

5.1 Datasets

We use two real-world rumor datasets, Twitter (Ma et al., 2017) and Weibo (Ma et al., 2016), to evaluate the SINCon approach. These datasets are sourced from two popular social media platforms—Twitter and Weibo. The Twitter dataset consists of English rumor datasets with conversation threads in tweets, providing a rich context for analysis. On the other hand, the Weibo dataset comprises Chinese rumor datasets with a similar composition structure. These datasets are annotated with two labels: Rumor and Non-Rumor, which are used for the binary classification of rumors and non-rumors. Detailed statistics for both datasets are provided in Appendix A.5.

We employ two metrics to validate the effectiveness of the proposed method: accuracy under attack (AUA.) and Accuracy (ACC.). Note that the higher the AUA. is, the more successful the defense method is. In contrast, a low ACC indicates a reduced performance of the rumor detector after the attack. Our primary goal is to evaluate the performance of SINCon on both clean data and data subjected to Message Injection Attacks. therefore, we take the ACC and AUA. as our primary metrics.

5.2 Settings

In our preliminary experiments, we employed the state-of-the-art Message injection attack, i.e., HMIA-LLM (Luo et al., 2024), to attack four MPT-based state-of-the-art rumor detectors:

- **BiGCN** (Bian et al., 2020): A GNN-based rumor detection model utilizing the Bi-directional propagation structure.
- **GACL** (Sun et al., 2022): A GNN-based model using adversarial and contrastive learning, which can not only encode the global propagation structure, but also resist noise and adversarial samples, and captures the event invariant features by utilizing contrastive learning.
- **GARD** (Tao et al., 2024a): A rumor detection model introduces self-supervised semantic evolvement learning to facilitate the acquisition of more transferable and robust representations.

We simulated two attack scenarios for defense: one where the attacker uses the same model for both generating the attack content and launching the attack, and another where the attacker employs a surrogate model to generate the attack content, then to attack the target model (i.e., the model generating the attack content is not necessarily the same as the target model). We conducted experiments on both the standard rumor detection model (Normal) and the model enhanced with SINCon (w/ SINCon) to evaluate ACC. and AUA..

In executing HMIA-LLM, We followed the settings from the original study (Luo et al., 2024), employing ChatGPT (gpt-3.5-turbo) to generate malicious messages, with a root-centric homophily threshold λ = 0.35 and a budget of Δ = 50. The dataset is partitioned with a 40% test and 60% training ratio, randomly shuffled to avoid order bias, balanced for label distribution using a stratified approach, and divided into five folds for cross-validation, with each fold serving as the test set once. Our experiments were conducted on a remote machine server with 1 NVIDIA RTX 3090 (24G) GPU. We set α_1=1e-5, α_2=1e-2 for Twitter, and α_1=1e-4, α_2=1e-4 for Weibo.

5.3 Overall Performance

The experimental results Table1 shows that SINCon significantly enhances the robustness of

Surrogate Model	Target Model	Method	Twitter		Weibo	
			AUA.	ACC.	AUA.	ACC.
BiGCN	BiGCN	Normal	0.7604	0.8979	0.6457	**0.9137**
		w/ SINCon	**0.8833**	**0.9021**	**0.8697**	0.9089
	GACL	Normal	0.6250	**0.9000**	0.7325	0.8999
		w/ SINCon	**0.8458**	0.8750	**0.8930**	0.8999
	GARD	Normal	0.6417	**0.8854**	0.9096	**0.9258**
		w/ SINCon	**0.7750**	0.8729	**0.9237**	0.9232
GACL	BiGCN	Normal	0.8417	**0.8979**	0.5779	**0.9153**
		w/ SINCon	**0.8729**	0.8708	**0.7865**	0.8898
	GACL	Normal	0.5354	**0.8750**	0.4931	**0.9200**
		w/ SINCon	**0.8250**	0.8583	**0.8543**	0.9041
	GARD	Normal	0.6604	**0.8917**	0.9015	**0.9359**
		w/ SINCon	**0.8333**	0.8750	**0.9174**	0.9258
GARD	BiGCN	Normal	0.7792	**0.9000**	0.5646	**0.9110**
		w/ SINCon	**0.8333**	0.8917	**0.7969**	0.8898
	GACL	Normal	0.5667	**0.9042**	0.4995	**0.9142**
		w/ SINCon	**0.8500**	0.8875	**0.7797**	0.8898
	GARD	Normal	0.6854	**0.8917**	0.7188	**0.9306**
		w/ SINCon	**0.7708**	0.8792	**0.8231**	0.9168

Table 1: We compare model accuracy under attack (AUA.) and accuracy (ACC.). The **bold** values of AUA. and ACC. represent the strongest robustness and the highest accuracy, respectively. Normal refers to the standard rumor detection model, while w/ SINCon denotes the rumor detection model enhanced with SINCon.

MPT-based rumor detection models against LLM-generated message injection attacks.

As shown in Table1, when combined with other rumor detection models, SINCon only introduces a slight negative effect on the accuracy of clean (Normal) data. Our analysis indicates that the maximum decrease in accuracy is 2.55%, with some cases showing no decrease at all, and an average decline of 1.38%. Overall, SINCon results in a modest reduction in model accuracy on clean data, a drop that primarily stems from the introduction of augmented samples used to implement the contrastive learning regularization.

SINCon significantly enhances the robustness of the model. As shown in Table 1, across various rumor detection models, SINCon markedly improves performance when facing LLM-driven malicious message injection attacks, enabling the model to better resist adversarial perturbations. The analysis demonstrates that AUA achieves a maximum improvement of 36.12%, a minimum improvement of 1.59%, and an average improvement of 16.63%. This approach substantially strengthens the model's robustness in adversarial environments while maintaining high accuracy on clean data. Overall, the experimental results in Table 1 clearly demonstrate the exceptional effectiveness of SINCon in enhancing the model's resilience against LLM-generated malicious message injection attacks.

5.4 Ablation Study

5.4.1 Data Augmentation Operation

We conducted an ablation study to further explore the impact of the "Similarizing the Influence of Nodes" data augmentation operation on SINCon. Specifically, we replaced the data augmentation operations$t^{imp}(\cdot)$ and $t^{ump}(\cdot)$ in SINCon with a new data augmentation strategy that randomly masks nodes. This experiment was carried out using two different model combinations on both the Twitter and Weibo datasets. As shown in Table 2, the performance (ACC. and AUA.) of the influence-based data augmentation strategy significantly outperforms the random node masking approach on

Surrogate Model	Target Model	Method	Twitter		Weibo	
			AUA.	ACC.	AUA.	ACC.
BiGCN	BiGCN	SWICon	**0.8833**	**0.9021**	**0.8697**	**0.9089**
		w/ random	0.8178	0.8204	0.8019	0.8427
	GACL	SWICon	**0.8458**	**0.8750**	**0.8930**	**0.8999**
		w/ random	0.8146	0.7646	0.8120	0.8056

Table 2: Experimental results of SINCon with different data augmentation operation. w/ random means the augmentations of each MPT are sampled randomly rather than based on attributions.

both datasets. These results provide additional evidence that the method of similarizing the influence of nodes plays a crucial role in enhancing the robustness of SINCon in rumor detection models, effectively counteracting the impact of adversarial attacks. This finding further solidifies our hypothesis that balancing node influence improves the model's overall performance and resilience.

5.4.2 Hyperparameter α_1

α_1 affects the weight of the supervised loss for the augmented data. In this experiment, we performed a sensitivity analysis on the hyperparameter α_1. Specifically, we adjusted the value of α_1 and compared the model's performance under different settings. The experiments were conducted using BiGCN surrogate and target models for ablation studies. Figure 3 shows the trend of changes in ACC. and AUA. values of the model on the Twitter and Weibo datasets under different α_1 values.

As shown in the experimental results, adjusting α_1 has a certain impact on the performance of SINCon, both on the Twitter and Weibo datasets. The ACC. in Normal and Normal+SINCon are similar, but when α_1 is too large or too small, Normal+SINCon performance slightly decreases. Similarly, for AUA., the performance of Normal+SINCon drops when α_1 is extreme. This is because the model tends to overfit the original MPT during the training process.

5.4.3 Hyperparameter α_2

This weight affects the result of SINCon by affecting the weight of contrastive loss in the total loss. In this experiment, we conducted a sensitivity analysis of the hyperparameter α_2 to assess its impact on model performance. The experiment used BiGCN-based surrogate and target models, and by adjusting the value of α_2, we observed the variations in model performance (ACC. and AUA.) on the Twitter and Weibo datasets. From the experimental results shown in Figure 4, it is evident that α_2 has a noticeable impact on model performance. α_2 affects the weight of the $\mathcal{L}_{\text{SINCon}}$ in the total loss function. First, regarding ACC, both excessively large and small values of α_2 result in a certain degree of performance degradation. Compared to ACC., α_2 has a more significant effect on AUA.. The results indicate that by balancing the influence of nodes in the MPTs, SINCon greatly enhances robustness against LLM-based message injection attacks.

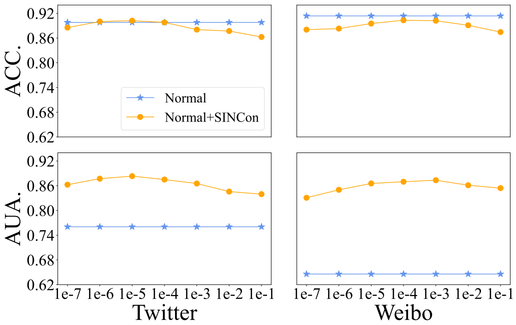

Figure 3: Sensitivity analysis of hyperparameters α_1. Experiments conducted with both the Surrogate Model and Target Model as BiGCN.

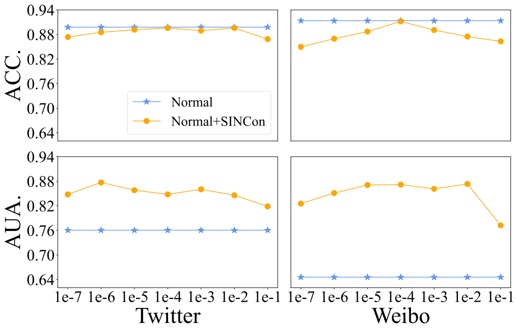

Figure 4: Sensitivity analysis of hyperparameters α_2. Experiments conducted with both the Surrogate Model and Target Model as BiGCN.

6 Conclusion

In this paper, we proposed SINCon, a defense mechanism for enhancing the robustness of MPT-based rumor detection models against adversarial message injection attacks. By leveraging contrastive learning, SINCon ensures that both important and unimportant nodes exert more uniform influence on the model's predictions, effectively mitigating the impact of localized perturbations caused by malicious message injections. Through extensive experiments on Twitter and Weibo datasets, we demonstrated that SINCon significantly improves the model's resilience to LLM-driven attacks while maintaining high classification accuracy on clean data.

7 Limitations

SINCon significantly enhances the performance of rumor detection models against LLM-driven message injection attacks, though at the cost of a slight decline in performance on clean data(an average of 1.48%). Future research could further explore how to optimize data augmentation strategies and loss function design, aiming to improve the model's defensive robustness while maintaining high accuracy on clean data. Moreover, this paper primarily focuses on LLM-driven malicious message injection attacks. However, in real-world environments, the methods of attack are becoming increasingly diverse. Future research should further examine the performance of SINCon against other types of attacks and explore more generalizable defense mechanisms.

8 Acknowledgments

This work is jointly sponsored by National Natural Science Foundation of China (62372454, 62236010, 62141608, 62206291).

References

Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 549–556.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Zeyu Cui, Zekun Li, Shu Wu, Xiaoyu Zhang, Qiang Liu, Liang Wang, and Mengmeng Ai. 2022. Dygcn: Efficient dynamic graph embedding with graph convolutional network. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):4635–4646.

Shujie Deng, Honghua Dong, and Xujie Si. Enhancing and evaluating logical reasoning abilities of large language models. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*.

Junyuan Fang, Haixian Wen, Jiajing Wu, Qi Xuan, Zibin Zheng, and K Tse Chi. 2024. Gani: Global attacks on graph neural networks via imperceptible node injections. *IEEE Transactions on Computational Social Systems*.

Haisong Gong, Weizhi Xu, Shu Wu, Qiang Liu, and Liang Wang. 2024. Heterogeneous graph reasoning for fact checking over texts and tables. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 100–108.

Lukas Gosch, Simon Geisler, Daniel Sturm, Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. 2024. Adversarial training for graph neural networks: Pitfalls, solutions, and new directions. *Advances in Neural Information Processing Systems*, 36.

Yuelin Hu, Futai Zou, Jiajia Han, Xin Sun, and Yilei Wang. 2024. Llm-tikg: Threat intelligence knowledge graph construction utilizing large language model. *Computers & Security*, 145:103999.

Kung-Hsiang Huang, Kathleen McKeown, Preslav Nakov, Yejin Choi, and Heng Ji. 2022. Faking fake news for real fake news detection: Propaganda-loaded training data generation. *arXiv preprint arXiv:2203.05386*.

Wenlong Huang, Fei Xia, Dhruv Shah, Danny Driess, Andy Zeng, Yao Lu, Pete Florence, Igor Mordatch, Sergey Levine, Karol Hausman, et al. 2024. Grounded decoding: Guiding text generation with grounded models for embodied agents. *Advances in Neural Information Processing Systems*, 36.

Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable rumor detection in microblogs by attending to user interactions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8783–8790.

Sarah Kreps, R Miles McCain, and Miles Brundage. 2022. All the news that's fit to fabricate: Ai-generated text as a tool of media misinformation. *Journal of experimental political science*, 9(1):104–117.

Taeyoon Kwon, Kai Tzu-iunn Ong, Dongjin Kang, Seungjun Moon, Jeong Ryong Lee, Dosik Hwang, Beomseok Sohn, Yongsik Sim, Dongha Lee, and Jinyoung Yeo. 2024. Large language models are clinical reasoners: Reasoning-aware diagnosis framework with prompt-generated rationales. In *Proceedings of* *the AAAI Conference on Artificial Intelligence*, volume 38, pages 18417–18425.

Guoyi Li, Die Hu, Zongzhen Liu, Xiaodan Zhang, and Honglei Lyu. 2025. Semantic reshuffling with llm and heterogeneous graph auto-encoder for enhanced rumor detection. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8557–8572.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2024. Pre-trained language models for text generation: A survey. *ACM Computing Surveys*, 56(9):1–39.

Qing Li, Ziyue Wang, and Zehao Li. 2023. Pagcl: An unsupervised graph poisoned attack for graph contrastive learning model. *Future Generation Computer Systems*, 149:240–249.

Guofan Liu, Jinghao Zhang, Qiang Liu, Junfei Wu, Shu Wu, and Liang Wang. 2024a. Uni-modal event-agnostic knowledge distillation for multimodal fake news detection. *IEEE Transactions on Knowledge and Data Engineering*.

Qiang Liu, Junfei Wu, Shu Wu, and Liang Wang. 2024b. Out-of-distribution evidence-aware fake news detection via dual adversarial debiasing. *IEEE Transactions on Knowledge and Data Engineering*.

Tianrui Liu, Qi Cai, Changxin Xu, Bo Hong, Fanghao Ni, Yuxin Qiao, and Tsungwei Yang. 2024c. Rumor detection with a novel graph neural network approach. *arXiv preprint arXiv:2403.16206*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. 2023. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.

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

Jason Lucas, Adaku Uchendu, Michiharu Yamashita, Jooyoung Lee, Shaurya Rohatgi, and Dongwon Lee. 2023. Fighting fire with fire: The dual role of llms in crafting and detecting elusive disinformation. *arXiv preprint arXiv:2310.15515*.

Yifeng Luo, Yupeng Li, Dacheng Wen, and Liang Lan. 2024. Message injection attack on rumor detection under the black-box evasion setting using large language model. In *Proceedings of the ACM on Web Conference 2024*, pages 4512–4522.

Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting rumors from microblogs with recurrent neural networks.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2017. Detect rumors in microblog posts using propagation structure via kernel learning. Association for Computational Linguistics.

Aleksander Mądry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *stat*, 1050(9).

Bhaye Malhotra and Dinesh Kumar Vishwakarma. 2020. Classification of propagation path and tweets for rumor detection using graphical convolutional networks and transformer based encodings. In *2020 IEEE sixth international conference on multimedia big data (BigMM)*, pages 183–190. IEEE.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Yikang Pan, Liangming Pan, Wenhu Chen, Preslav Nakov, Min-Yen Kan, and William Yang Wang. 2023. On the risk of misinformation pollution with large language models. *arXiv preprint arXiv:2305.13661*.

Yun-Zhu Song, Yi-Syuan Chen, Yi-Ting Chang, Shao-Yu Weng, and Hong-Han Shuai. 2021. Adversary-aware rumor detection. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1371–1382.

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

Xin Sun, Liang Wang, Qiang Liu, Shu Wu, Zilei Wang, and Liang Wang. 2024a. Dive: subgraph disagreement for graph out-of-distribution generalization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2794–2805.

Yanshen Sun, Jianfeng He, Limeng Cui, Shuo Lei, and Chang-Tien Lu. 2024b. Exploring the deceptive power of llm-generated fake news: A study of real-world detection challenges. *arXiv preprint arXiv:2403.18249*.

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

Xiang Tao, Mingqing Zhang, Qiang Liu, Shu Wu, and Liang Wang. 2024b. Out-of-distribution rumor detection via test-time adaptation. *arXiv preprint arXiv:2403.17735*.

Adaku Uchendu, Jooyoung Lee, Hua Shen, Thai Le, Dongwon Lee, et al. 2023. Does human collaboration enhance the accuracy of identifying llm-generated deepfake texts? In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 11, pages 163–174.

Karthik Valmeekam, Matthew Marquez, Sarath Sreedharan, and Subbarao Kambhampati. 2023. On the planning abilities of large language models-a critical investigation. *Advances in Neural Information Processing Systems*, 36:75993–76005.

Junfei Wu, Qiang Liu, Weizhi Xu, and Shu Wu. 2022. Bias mitigation for evidence-aware fake news detection by causal intervention. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2308–2313.

Junfei Wu, Weizhi Xu, Qiang Liu, Shu Wu, and Liang Wang. 2023. Adversarial contrastive learning for evidence-aware fake news detection with graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*.

Zhiyuan Wu, Dechang Pi, Junfu Chen, Meng Xie, and Jianjun Cao. 2020. Rumor detection based on propagation graph neural network with attention mechanism. *Expert systems with applications*, 158:113595.

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

Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. 2023. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345*.

Xuhai Xu, Bingsheng Yao, Yuanzhe Dong, Saadia Gabriel, Hong Yu, James Hendler, Marzyeh Ghassemi, Anind K Dey, and Dakuo Wang. 2024. Mental-llm: Leveraging large language models for mental health prediction via online text data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 8(1):1–32.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Shaochen Zhong, Bing Yin, and Xia Hu. 2024. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *ACM Transactions on Knowledge Discovery from Data*, 18(6):1–32.

Zhengli Zhai, Penghui Li, and Shu Feng. 2023. State of the art on adversarial attacks and defenses in graphs. *Neural Computing and Applications*, 35(26):18851–18872.

Pengwei Zhan, Jing Yang, He Wang, Chao Zheng, Xiao Huang, and Liming Wang. 2023. Similarizing the influence of words with contrastive learning to defend word-level adversarial text attack. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7891–7906.

Chenhan Zhang, Shiyao Zhang, James JQ Yu, and Shui Yu. 2023a. Sam: Query-efficient adversarial attacks against graph neural networks. *ACM Transactions on Privacy and Security*, 26(4):1–19.

Huaiwen Zhang, Xinxin Liu, Qing Yang, Yang Yang, Fan Qi, Shengsheng Qian, and Changsheng Xu. 2024a. T3rd: Test-time training for rumor detection on social media. In *Proceedings of the ACM on Web Conference 2024*, pages 2407–2416.

Jiliang Zhang and Chen Li. 2019. Adversarial examples: Opportunities and challenges. *IEEE transactions on neural networks and learning systems*, 31(7):2578–2593.

Kaiwei Zhang, Junchi Yu, Haichao Shi, Jian Liang, and Xiao-Yu Zhang. 2023b. Rumor detection with diverse counterfactual evidence. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3321–3331.

Mengmei Zhang, Xiao Wang, Chuan Shi, Lingjuan Lyu, Tianchi Yang, and Junping Du. 2023c. Minimum topology attacks for graph neural networks. In *Proceedings of the ACM Web Conference 2023*, pages 630–640.

Mingqing Zhang, Haisong Gong, Qiang Liu, Shu Wu, and Liang Wang. 2024b. Breaking event rumor detection via stance-separated multi-agent debate. *arXiv preprint arXiv:2412.04859*.

Zoie Zhao, Sophie Song, Bridget Duah, Jamie Macbeth, Scott Carter, Monica P Van, Nayeli Suseth Bravo, Matthew Klenk, Kate Sick, and Alexandre LS Filipowicz. 2023. More human than human: Llm-generated narratives outperform human-llm interleaved narratives. In *Proceedings of the 15th Conference on Creativity and Cognition*, pages 368–370.

Andrew Zhu, Lara Martin, Andrew Head, and Chris Callison-Burch. 2023. Calypso: Llms as dungeon master's assistants. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 19, pages 380–390.

Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2024. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. *World Wide Web*, 27(5):58.

Xu Zou, Qinkai Zheng, Yuxiao Dong, Xinyu Guan, Evgeny Kharlamov, Jialiang Lu, and Jie Tang. 2021. Tdgia: Effective injection attacks on graph neural networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2461–2471.

Threshold	0		0.05		0.1		0.15		0.2		0.25	
	ACC	AUA	ACC	AUA	ACC	AUA	ACC	AUA	ACC	AUA	ACC	AUA
Twitter	0.875	0.8729	0.8875	0.8775	0.9021	0.8833	0.9	0.8788	0.9	0.8775	0.8979	0.8688
Weibo	0.9137	0.7604	0.9068	0.8077	0.9089	0.8697	0.9068	0.8697	0.9021	0.8479	0.893	0.8425

Table 3: Performance comparison at different thresholds on Twitter and Weibo datasets

Surrogate Model	Target Model	Method	Twitter		Weibo	
			ACC.	AUA.	ACC.	AUA.
BiGCN	BiGCN	SINCon	0.9021	0.8833	0.9089	0.8697
BiGCN	BiGCN	w/o contrastive loss	0.8896	0.8479	0.8273	0.7500
BiGCN	GACL	SINCon	0.8750	0.8458	0.8999	0.8930
BiGCN	GACL	w/o contrastive loss	0.8500	0.8104	0.8141	0.8077
BiGCN	GARD	SINCon	0.8729	0.7750	0.9232	0.9237
BiGCN	GARD	w/o contrastive loss	0.8583	0.7542	0.8200	0.8274

Table 4: Ablation Results of Contrastive Loss on Twitter and Weibo Datasets, “w/o contrastive loss” indicates the SINCon variant without the contrastive learning loss.

A Appendix

A.1 system prompt p

Instruction: Your mission is to construct a sentence that bears the least semantic similarity to the user’s inputs while maintaining a similar overarching topic. Cosine similarity will be used to evaluate the dissimilarity.

A.2 iterative prompting p'

Instruction: The similarity between the generated sentence and the input sentence is $\{\text{sim}_{\text{root}}(G)\}$.
Please generate a new sentence.

A.3 Datasets

Statistics	**Twitter**	**Weibo**
Users#	491229	2746818
Posts#	1101985	3805656
MPTs#	992	4664
Rumors#	498	2313
Non-Rumors#	494	2351
Avg. time length/MPT	1582.6 Hours	2460.7 Hours
Avg # of posts/MPT	1111	816
Max # of posts/MPT	62827	59318
Min # of posts/MPT	10	10
Language	English	Chinese

Table 5: Statistics of the datasets.

A.4 Sensitivity analysis under different masking thresholds

we have conducted sensitivity tests with different thresholds , using both the Surrogate Model and Target Model as BiGCN. The experimental results are summarized in the Table 3. As shown in the results, the model achieves optimal performance when the selected proportion is 10%. We have now included these additional experiments and analyses in the revised manuscript to enhance the completeness of our study.

A.5 Ablation Study on Contrastive Learning

The results show the performance variations when the contrastive loss is removed in Table 4. As indicated by the results, the removal of the contrastive loss leads to varying degrees of performance degradation across the experiments. These findings demonstrate that by balancing the influence of nodes within the MPTs, SINCon significantly enhances the model’s robustness against LLM-based message injection attacks.