# CPNS: Curriculum Learning-Guided Progressive Key Node Selection for Fake News Detection

**Anonymous authors**

## Abstract

Selecting valuable nodes from the social propagation graph is crucial for fake news detection. Previous work mainly evaluates node importance based on propagation depth, then constructs propagation subgraphs to mitigate noise interference. However, they still suffer from two limitations: (1) propagation subgraphs make signals from both valuable and noisy nodes smoother; (2) affected by depth, deep nodes are easily ignored. To address these issues, we propose **C**urriculum Learning-guided **P**rogressive Key **N**ode **S**election for Fake News Detection (CPNS) to select key propagation nodes. Specifically, we propose a curriculum learning-guided progressive propagation strategy to select deep nodes. Moreover, we weight nodes based on properties, thereby distilling informative knowledge from selected nodes. Finally, we introduce multi-frequency consistency learning. It jointly models low-frequency propagation consensus and high-frequency local differences, thereby avoiding excessive smoothing of both noisy and valuable nodes. Experimental results demonstrate that our approach achieves state-of-the-art performance on two benchmark datasets, improving Macro-$F_1$ by 1.54% and 2.15%, respectively.

## 1 Introduction

On social platforms such as Twitter and Weibo, news propagation graphs contain numerous user nodes (Chen et al., 2026). However, a substantial number of these propagation nodes are noisy nodes, including social bots (Zhang et al., 2024) and malicious users (Zhang et al., 2025), among others. Such noisy nodes interfere with effective propagation representations. This reduces news detection accuracy (ACC) and threatens social stability (Yin et al., 2025; Zhu et al., 2026). Therefore, effectively selecting valuable nodes and constructing propagation graphs are crucial for fake news detection (Yin et al., 2026).

Selecting valuable nodes from noisy propagation graphs is important for fake news detection (Zhang et al., 2025; Wei et al., 2021). Previous work selects valuable nodes through soft node weighting (Guo et al., 2025; Hu et al., 2026; Wei et al., 2021) or hard structure selection (Zhang et al., 2025; Liu et al., 2026b), and then learns node features. However, these methods only learn individual node representations, while ignoring relationships between nodes (Bian et al., 2020; Dou et al., 2021). Recent studies construct propagation subgraphs (Yin et al., 2025; Peng et al., 2026; Gong et al., 2026) to select valuable nodes and model their relationships (Hu et al., 2026). Specifically, such methods (Bian et al., 2020; Yang et al., 2020; Wei et al., 2021) use neighborhood aggregation, which integrates propagation representations from neighboring nodes (Guo et al., 2025). However, neighborhood aggregation makes node representations smooth, thereby making different nodes less distinguishable. Such an effect is more pronounced for nodes deeper in propagation paths. Deep valuable nodes are assigned lower weights, and are misclassified as noisy nodes.

To validate these issues, we conduct two experiments: (1) We analyze node similarity in subgraphs selected by KPG (Zhang et al., 2025) and HCSL (Liu et al., 2026a) (Table 1). Concretely, the weighted squared difference between nodes continuously decreases during subgraph aggregation. This indicates that neighborhood aggregation makes node representations increasingly similar. Such aggregation further aggravates over-smoothing in propagation graphs. (2) We further compare deep and shallow nodes sampled from PolitiFact and GossipCop (Figure 1). In fact, both types of nodes contain similar amounts of valuable information, suggesting that propagation depth cannot reflect node information value. Therefore, subgraph construction should not reduce node contributions solely because of greater propagation depth.



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To tackle the above challenges, we propose CPNS to progressively identify valuable nodes and preserve discriminative propagation information. Unlike methods that only rely on propagation depth or construct key propagation subgraphs through hard selection, CPNS draws on the progressive optimization principle of curriculum learning. It transforms propagation depth into structural information exploited in stages. This enables more effective selection of valuable nodes from noisy propagation graphs.

Table 1: Weighted squared difference between adjacent nodes on PolitiFact and GossipCop. Lower values indicate smoother node representations, which are underlined. $K$ denotes the propagation round.

| | PolitiFact | | GossipCop | |
|---|---|---|---|---|
| $K$ | KPG | HCSL | KPG | HCSL |
| 0 | 0.90744 | 0.91398 | 0.79393 | 0.79406 |
| 1 | 0.17592 | 0.17665 | 0.16482 | 0.16476 |
| 2 | 0.04181 | 0.04172 | 0.04601 | 0.04597 |
| 3 | 0.01125 | 0.01118 | 0.01457 | 0.01457 |
| 4 | 0.00318 | 0.00313 | 0.00503 | 0.00496 |
| 5 | 0.00094 | 0.00091 | 0.00193 | 0.00181 |
| 6 | 0.00029 | 0.00027 | 0.00077 | 0.00071 |

Specifically, we design a node evidence enhancement module. It combines propagation representations to evaluate the evidence value and reliability of nodes. Then, we design a propagation-aware node selection module. It gradually introduces information from different propagation depths through curriculum learning. The multi-branch design reduces the incorrect association between propagation depth and node value. Finally, we introduce a multi-frequency learning module. It decomposes propagation representations into low-frequency and high-frequency signals. This reduces the loss of valuable information caused by over-smoothing during propagation denoising. The resulting propagation representations are fused with news semantics to predict news veracity.

The main contributions of this paper are summarized as follows:

(1) We are the first to investigate whether propagation depth indicates node information value and reveal that deep nodes can preserve valuable information during noise suppression.

(2) Our propagation-aware node selection combines the progressive strategy of curriculum learning with propagation depth. It effectively retains propagation graph information. Meanwhile, multi-frequency learning mitigates discriminative information loss during noise suppression.

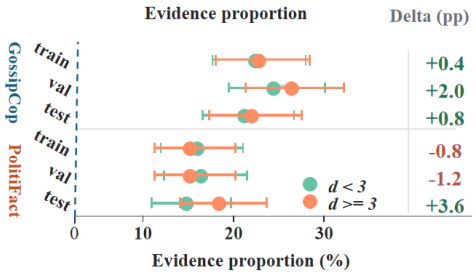


Figure 1: Comparison of valuable nodes at different propagation depths in GossipCop and PolitiFact.

(3) Macro-$F_1$ improvements of 1.54% and 2.15% are achieved on the large-sample dataset GossipCop and the small-sample dataset PolitiFact, respectively, significantly outperforming state-of-the-art baseline methods.

## 2 Related Work

### 2.1 Propagation Structure Learning for Fake News Detection

Propagation patterns provide valuable structural evidence for fake news detection (Bian et al., 2020; Wei et al., 2021). Early studies mainly model how user responses diffuse along propagation structures (Bian et al., 2020). Bi-GCN (Bian et al., 2020) models top-down propagation and bottom-up dispersion with bidirectional graph convolution. Subsequent methods enrich propagation modeling with uncertainty and hierarchical semantics (Liu et al., 2026a; Wei et al., 2021). EBGCN (Wei et al., 2021) captures uncertainty from unreliable propagation relations through Bayesian modeling. HCSL (Liu et al., 2026a) integrates curriculum learning with hierarchical semantic modeling. These methods progressively improve structural representation quality across propagation levels (Bian et al., 2020; Liu et al., 2026a; Wei et al., 2021). However, stronger structure modeling does not directly resolve noisy propagation information (Zhu et al., 2024). This limitation motivates dedicated propagation

structure learning and refinement methods (Wu and Hooi, 2023; Cui and Jia, 2024; Yin et al., 2024; Zhu et al., 2024).

### 2.2 Node Selection in Propagation Graphs

Propagation graphs often contain noisy nodes, which interfere with effective propagation modeling (Zhang et al., 2025; Zhu et al., 2024). Therefore, existing methods select valuable nodes or structures to reduce noise (Yang et al., 2022; Zhang et al., 2025; Cui and Jia, 2024; Gong et al., 2026). RAGCL (Cui and Jia, 2024) estimates node importance using structural centrality for adaptive node dropping. KPG (Zhang et al., 2025) iteratively selects valuable nodes to generate key propagation graphs. CSDA (Gong et al., 2026) learns node and edge masks to extract invariant propagation subgraphs. SureFact (Yang et al., 2022) identifies discriminative local structures through subgraph reasoning. These methods rely on explicit selection to retain valuable propagation evidence. However, hard selection may overlook deep nodes with underestimated importance. Removing nodes may also discard potentially valuable information. Moreover, subsequent message passing can smooth valuable and noisy node signals (Zhu et al., 2024). Soft methods instead adjust propagation contributions without directly removing structures (Wu and Hooi, 2023; Zhu et al., 2024). DECOR (Wu and Hooi, 2023) corrects noisy relations by adjusting their contributions during message propagation. This reduces the influence of unreliable relations while preserving the original graph structure. In contrast, PSGT (Zhu et al., 2024) suppresses task-irrelevant attention links during feature aggregation. However, they mainly refine relations rather than progressively selecting valuable nodes. Our CPNS progressively selects valuable nodes at greater propagation depths. It then weights selected nodes according to their properties. Multi-frequency consistency preserves propagation consensus and local differences across selected nodes.

## 3 Method

The overall framework of CPNS is illustrated in Figure 2. It first constructs morphology-conditioned candidate nodes and obtains label-isolated evidence estimates from a large language model (LLM). These estimates produce bounded node weights that are shared by the subsequent graph readouts. The propagation encoder then exposes edges progressively with a depth curriculum and uses node-level routing to estimate propagation uncertainty. Finally, the encoder output is decomposed into a main view, a local low-frequency view, and a high-frequency residual view. The latter two views are used only during training, whereas the main view is fused with the source-news representation for graph-level veracity prediction.

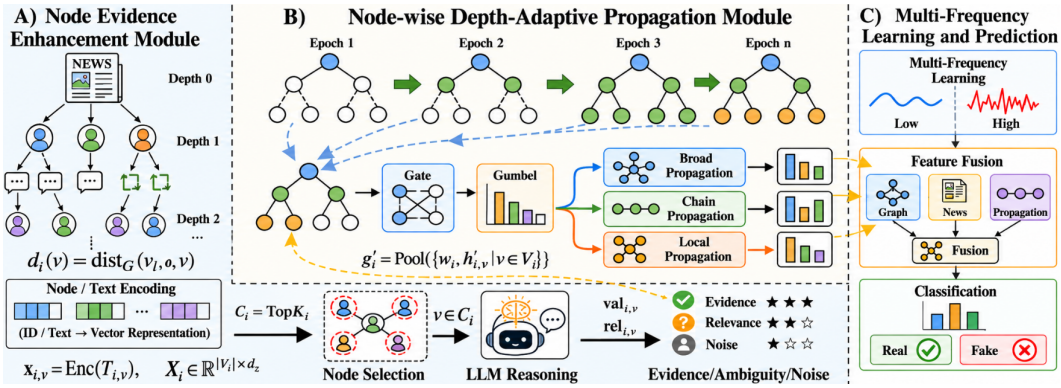


Figure 2: Overall framework of CPNS. It consists of three key modules: A) feature extraction based on propagation and node evidence assessment, B) curriculum learning-guided propagation representation with node-level multi-branch routing, and C) multi-frequency training constraints followed by main-view root-aware prediction.

### 3.1 Task Definition

We formulate fake news detection as a graph-level veracity prediction task over directed attributed propagation graphs. Specifically, each news item is represented as

$$G_i = (V_i, E_i, \mathbf{X}_i), \tag{1}$$

where $V_i$ is the node set, $E_i$ is the directed propagation-edge set, and $\mathbf{X}_i$ contains the text features of all nodes. The source-news node is $v_i^0 \in V_i$, and $y_i \in \{0, 1\}$ is the graph-level veracity label. For any node $v \in V_i$, $d_i(v)$ denotes its shortest-path distance from $v_i^0$, with $d_i(v_i^0) = 0$. The graph classifier $f_\theta$ outputs the two-class probability vector $\hat{\mathbf{y}}_i = (\hat{y}_{i,0}, \hat{y}_{i,1})^\top$. Our goal is to learn $f_\theta$ that predicts $y_i$ from $G_i$.

### 3.2 Propagation Graph Modeling and Node Evidence Enhancement

The shared text encoder maps node $v$ to $\mathbf{x}_{i,v} \in \mathbf{R}^{d_x}$, where $d_x$ is the feature dimension, and stacks all node features as

$$\mathbf{X}_i = [\mathbf{x}_{i,v}]_{v \in V_i} \in \mathbf{R}^{|V_i| \times d_x}. \tag{2}$$

We summarize propagation morphology using a deterministic descriptor function $\phi_{\text{morph}}$:

$$\mathbf{m}_i = [m_{i,1}, \ldots, m_{i,K}]^\top = \phi_{\text{morph}}(G_i) \in \mathbf{R}^K, \tag{3}$$

where $K$ denotes the number of morphology descriptors ($K = 10$ in our implementation), and $m_{i,1:K}$ respectively encode node count, maximum depth, depth-distribution entropy, maximum out-degree, hub depth, normalized maximum out-degree, first-level branch imbalance, maximum subtree fraction, deep-node ratio, and deep-burst score. We select the top-$k$ nodes according to $\mathbf{m}_i$. Details of node selection are provided in Appendix A.

For each $v \in \mathcal{C}_i$, LLM receives the root text, morphology summary, candidate text, and local parent–child context, without $y_i$ or a graph-level Real/Fake request. It returns $j_{i,v} \in \{\text{evidence}, \text{noise}, \text{ambiguous}\}$, $\text{val}_{i,v}, \text{rel}_{i,v} \in [0, 1]$, and a rationale, as shown in Figure 3.

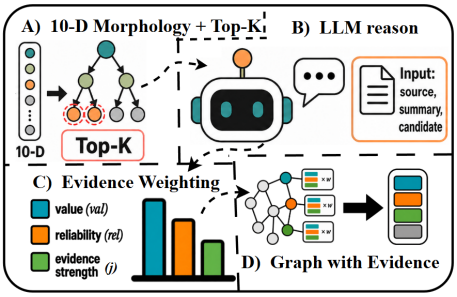


Figure 3: Morphology-conditioned candidate selection and LLM evidence weighting.

For completeness, define $\text{Clamp}(x, a, b) = \min(\max(x, a), b)$. We set the evidence score and the node weight as

$$\begin{aligned} z_{i,v} &= \begin{cases} \text{Clamp}(\text{val}_{i,v}\text{rel}_{i,v}, 0, 1), & v \in \mathcal{C}_i, \\ 0, & v \notin \mathcal{C}_i, \end{cases} \\ w_{i,v} &= \begin{cases} \text{Clamp}(1 + \gamma_c + \gamma_l z_{i,v}, w_{\min}, w_{\max}), & v \in \mathcal{C}_i, \\ 1, & v \notin \mathcal{C}_i. \end{cases} \end{aligned} \tag{4}$$

Here $\gamma_c$ and $\gamma_l$ control the candidate prior and evidence contribution, respectively, with fixed bounds $w_{\min} \le 1 \le w_{\max}$. Candidate nodes are evidence-weighted, while non-candidates retain unit weight.

### 3.3 Propagation-Aware Node Selection

Let $t$ index an update stage and let $Q_t \in \mathbb{N}_0$ be the maximum propagation depth visible at that stage. Training follows a monotonically increasing depth curriculum: shallow relations are exposed first and deeper relations are introduced later. The curriculum masks message-passing edges only; the hidden states remain indexed by the complete node set $V_i$, so a deep node is not deleted or labelled as noise:

$$E_i^{(t)} = \{(u, v) \in E_i \mid d_i(u) \le Q_t,\ d_i(v) \le Q_t\}. \tag{5}$$

For layer-$l$ states, let $d_h$ denote the hidden dimension and define $\mathbf{H}_i^{(l)} = [\mathbf{h}_{i,v}^{(l)}]_{v \in V_i} \in \mathbb{R}^{|V_i| \times d_h}$. A node-level gate mixes $K$ parallel GraphSAGE branches. With gating logit $a_{i,v,k}^{(l)}$, Gumbel noise $\epsilon_{i,v,k}^{(l)}$,

and temperature $\tau > 0$, the branch probabilities are

$$p_{i,v,k}^{(l)} = \frac{\exp\Big(\Big(a_{i,v,k}^{(l)} + \epsilon_{i,v,k}^{(l)}\Big)/\tau\Big)}{\sum_{k'=1}^{K} \exp\Big(\Big(a_{i,v,k'}^{(l)} + \epsilon_{i,v,k'}^{(l)}\Big)/\tau\Big)}. \tag{6}$$

The $k$th branch is denoted by $\mathcal{F}_k^{(l)}$, and $\sigma$ is the element-wise activation function. The routed node update is

$$\mathbf{h}_{i,v}^{(l+1)} = \sigma\left(\sum_{k=1}^{K} p_{i,v,k}^{(l)} \left[\mathcal{F}_k^{(l)}\Big(\mathbf{H}_i^{(l)}, E_i^{(t)}\Big)\right]_v\right). \tag{7}$$

Routing adapts aggregation without deterministic depth-based pruning. Its branch-selection uncertainty is measured by the layer-averaged Shannon entropy

$$u_i(v) = -\frac{1}{L}\sum_{l=1}^{L}\sum_{k=1}^{K} p_{i,v,k}^{(l)} \log p_{i,v,k}^{(l)}, \tag{8}$$

where $L$ is the number of propagation layers. This is the unnormalized entropy and satisfies $u_i(v) \in [0, \log K]$; the coefficients below therefore also balance the depth and entropy scales.

Let $d_i^{\max} = \max_{v \in V_i} d_i(v)$ and define the zero-safe normalized depth $\widetilde{d}_i(v) = d_i(v)/\max(d_i^{\max}, 1)$. We define $B$ as the set of graph indices in the current mini-batch, so that $i \in B$ indexes a graph used in the current update. Graph difficulty combines structural depth and routing uncertainty, and the resulting softmax determines the sample weight:

$$q_i(t) = \frac{1}{|V_i|}\sum_{v \in V_i}\Big[\alpha \widetilde{d}_i(v) + \beta u_i(v)\Big], \qquad \omega_i(t) = \frac{\exp(q_i(t))}{\sum_{i' \in B}\exp(q_{i'}(t))}. \tag{9}$$

Here $\alpha, \beta \geq 0$ and $\alpha + \beta = 1$; $\omega_i(t)$ scales the supervised graph loss for graph $i$ at stage $t$.

### 3.4 Multi-Frequency Consistency Learning and Prediction

Let $\mathbf{h}_{i,v} \in \mathbf{R}^{d_h}$ denote the final node representation produced by the curriculum-controlled encoder. Let $\mathcal{N}_i(v)$ denote the transmission neighborhood used by the local frequency filter. To make the local mean defined also for an empty neighborhood, we use the node state itself as the mean when $\mathcal{N}_i(v) = \varnothing$; otherwise the mean is computed over $\mathcal{N}_i(v)$. We index the three views, where $m$, $l$, and $h$ denote the main, low-frequency, and high-frequency views, respectively. The three node views are

$$\begin{aligned}
\mathbf{h}_{i,v}^{m} &= \mathbf{h}_{i,v}, \\
\mathbf{h}_{i,v}^{l} &= \begin{cases} \dfrac{1}{|\mathcal{N}_i(v)|}\displaystyle\sum_{u \in \mathcal{N}_i(v)} \mathbf{h}_{i,u}, & |\mathcal{N}_i(v)| > 0, \\ \mathbf{h}_{i,v}, & |\mathcal{N}_i(v)| = 0, \end{cases} \\
\mathbf{h}_{i,v}^{h} &= \mathbf{h}_{i,v} - \mathbf{h}_{i,v}^{l}.
\end{aligned} \tag{10}$$

The low-frequency view captures local consensus, whereas the high-frequency view preserves deviations from that consensus. All three views share the evidence weight and the same element-wise max-pooling operator:

$$\mathbf{g}_i^q = \mathrm{MaxPool}\big(\{w_{i,v}\mathbf{h}_{i,v}^q \mid v \in V_i\}\big), \quad q \in \{m, l, h\}. \tag{11}$$

Here $q$ indexes the view. Each pooled representation satisfies $\mathbf{g}_i^q \in \mathbf{R}^{d_h}$. The low- and high-frequency graph representations are used only during training. We denote the implemented label-conditioned cross-view objective by $\mathcal{L}_{\text{NCE}}$. The multi-frequency consistency loss is defined as

$$\mathcal{L}_{\text{freq}} = \mathcal{L}_{\text{NCE}}\big(\mathbf{g}_i^m, \mathbf{g}_i^l, y_i\big) + \mathcal{L}_{\text{NCE}}\big(\mathbf{g}_i^m, \mathbf{g}_i^h, y_i\big). \tag{12}$$

Figure 4 illustrates the shared weighting and pooling operation. The exact temperature, projection, and positive-pair rule of $\mathcal{L}_{\text{NCE}}$ must match the released training configuration. At inference time, only the main view is retained and fused with the root-news embedding and morphology descriptor. Let $\mathbf{r}_i = \mathbf{x}_{i,v_i^0}$ denote the source-news embedding. The learnable map $\phi_{\text{root}}$ combines $\mathbf{r}_i$ with the main graph representation $\mathbf{g}_i^m$. We denote the resulting fused graph representation by $\mathbf{g}_i$. It is concatenated with $\mathbf{m}_i$ and passed to a linear classifier with parameters $\mathbf{W}_c$ and $\mathbf{b}_c$:

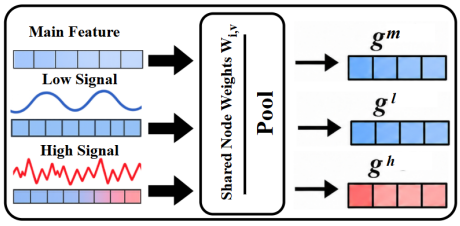


Figure 4: Shared node weighting and max pooling for the main, low-frequency, and high-frequency views.

$$\mathbf{g}_i = \phi_{\text{root}}(\mathbf{r}_i, \mathbf{g}_i^m) \,\|\, \mathbf{m}_i, \qquad \hat{\mathbf{y}}_i = \text{softmax}(\mathbf{W}_c \mathbf{g}_i + \mathbf{b}_c)\,. \tag{13}$$

Here $\|$ denotes vector concatenation, and $\hat{\mathbf{y}}_i$ is the resulting two-class probability vector. The weighted supervised loss is defined below, where CE denotes graph-level cross-entropy:

$$\mathcal{L}_{\text{sup}} = \sum_{i \in B} \omega_i(t)\, \text{CE}(\hat{\mathbf{y}}_i, y_i)\,. \tag{14}$$

Let $\mathcal{L}_{\text{reg}}$ denote the minimization-form gate regularizer implemented by the routing module. If the implementation produces a maximization-form entropy reward $\mathcal{L}_{\text{div}}$, we use $\mathcal{L}_{\text{reg}} = -\mathcal{L}_{\text{div}}$.

The final training objective is

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{sup}} + \lambda_1 \mathcal{L}_{\text{reg}} + \lambda_2 \mathcal{L}_{\text{freq}}. \tag{15}$$

Here, $\lambda_1, \lambda_2 \geq 0$ control the routing regularization and multi-frequency consistency terms, respectively.

# 4 Experiments

## 4.1 Datasets and Experimental Setting

This paper uses two widely used public datasets from FakeNewsNet (Shu et al., 2020) for evaluation: **GossipCop** and **PolitiFact** (Appendix B.1). The training configuration and backbone-selection details are reported in Appendix B.2. Detailed metric definitions are provided in Appendix B.3.

## 4.2 Baseline Methods

We compared CPNS with two categories of representative graph-structured fake news detection methods. **(1) Propagation representation methods:** Bi-GCN (Bian et al., 2020), UPFD (Dou et al., 2021), CGAT (Yang et al., 2020), EBGCN (Wei et al., 2021), FGDGNN (Guo et al., 2025), MTAS (Le-Pham et al., 2026), H2DFD (Peng et al., 2026), and GAAC (Hu et al., 2026). **(2) Structure denoising and key-information methods:** BREAK (Yin et al., 2025), KPG (Zhang et al., 2025), SEE-GraphMAE (Liu et al., 2026b), CSDA (Gong et al., 2026), and HCSL (Liu et al., 2026a). Detailed descriptions of these baselines are provided in Appendix B.4.

## 4.3 Experimental and Comparison Results

**(1) Overall experimental comparison**

Tables 2 and 3 compare CPNS with all baselines on two highly noisy datasets. CPNS achieves the best performance across all reported metrics. Notably, Macro-$F_1$ improves by 1.54% and 2.15% on the two datasets, respectively. These gains remain consistent despite different dataset scales. The performance gains stem from jointly addressing the two limitations identified above. Propagation-aware selection progressively introduces nodes across depths instead of uniformly suppressing deeper structures. Multi-frequency learning further preserves local distinctions weakened by repeated neighborhood aggregation. Together, these mechanisms produce reliable and discriminative propagation

Table 2: Overall performance comparison on GossipCop.

| Methods | ACC | Macro-$F_1$ | Precision | Recall |
|---|---|---|---|---|
| GAMC (Yin et al., 2024) | 0.8147 | 0.8143 | 0.8175 | 0.8148 |
| UPFD-GCN (Dou et al., 2021) | 0.8265 | 0.8262 | 0.8284 | 0.8265 |
| UPFD-GAT (Dou et al., 2021) | 0.8662 | 0.8657 | 0.8713 | 0.8663 |
| FGDGNN (Guo et al., 2025) | 0.8840 | 0.8838 | 0.8862 | 0.8840 |
| KPG (Zhang et al., 2025) | 0.8855 | 0.8854 | 0.8869 | 0.8856 |
| GAAC (Hu et al., 2026) | 0.8873 | 0.8873 | 0.8885 | 0.8874 |
| Bi-GCN (Bian et al., 2020) | 0.9315 | 0.9315 | 0.9320 | 0.9315 |
| MTAS (Le-Pham et al., 2026) | 0.9370 | 0.9370 | 0.9372 | 0.9370 |
| BREAK (Yin et al., 2025) | 0.9373 | 0.9373 | 0.9374 | 0.9373 |
| UPFD-SAGE (Dou et al., 2021) | 0.9386 | 0.9386 | 0.9390 | 0.9386 |
| H2DFD (Peng et al., 2026) | 0.9388 | 0.9388 | 0.9390 | 0.9389 |
| Semantic-GMAE (Liu et al., 2026b) | 0.9391 | 0.9391 | 0.9392 | 0.9391 |
| EBGCN (Wei et al., 2021) | 0.9469 | 0.9469 | 0.9470 | 0.9470 |
| CSDA (Gong et al., 2026) | 0.9618 | 0.9618 | 0.9623 | 0.9618 |
| CPNS (ours) | **0.9773** | **0.9772** | **0.9775** | **0.9772** |

Table 3: Overall performance comparison on PolitiFact.

| Methods | ACC | Macro-$F_1$ | Precision | Recall |
|---|---|---|---|---|
| Bi-GCN (Bian et al., 2020) | 0.8387 | 0.8360 | 0.8340 | 0.8397 |
| UPFD-GCN (Dou et al., 2021) | 0.8548 | 0.8517 | 0.8503 | 0.8536 |
| GAMC (Yin et al., 2024) | 0.8548 | 0.8482 | 0.8579 | 0.8429 |
| Semantic-GMAE (Liu et al., 2026b) | 0.8557 | 0.8547 | 0.8548 | 0.8547 |
| UPFD-SAGE (Dou et al., 2021) | 0.8579 | 0.8591 | 0.8605 | 0.8592 |
| KPG (Zhang et al., 2025) | 0.8699 | 0.8691 | 0.8705 | 0.8692 |
| BREAK (Yin et al., 2025) | 0.8702 | 0.8687 | 0.8688 | 0.8687 |
| GAAC (Hu et al., 2026) | 0.8705 | 0.8688 | 0.8700 | 0.8689 |
| CSDA (Gong et al., 2026) | 0.8710 | 0.8697 | 0.8688 | 0.8782 |
| MTAS (Le-Pham et al., 2026) | 0.8713 | 0.8701 | 0.8703 | 0.8701 |
| FGDGNN (Guo et al., 2025) | 0.8715 | 0.8704 | 0.8728 | 0.8706 |
| UPFD-GAT (Dou et al., 2021) | 0.8726 | 0.8710 | 0.8766 | 0.8715 |
| H2DFD (Peng et al., 2026) | 0.8832 | 0.8816 | 0.8792 | 0.8860 |
| EBGCN (Wei et al., 2021) | 0.8894 | 0.8876 | 0.8859 | 0.8899 |
| CPNS (ours) | **0.9194** | **0.9091** | **0.9108** | **0.9149** |

representations. This suggests that depth serves better as curriculum information than a selection criterion. Deep-node evidence can therefore contribute after stable shallow patterns are learned.

**(2) Comparison with baselines under noisy conditions.**

To validate model robustness to noise, we inject different levels of Gaussian noise into datasets. We compare our method with several strong baselines. Tables 4 and 5 compare CPNS with representative baselines under Gaussian noise. Under a noise level of 0.05, CPNS maintains the best performance on both datasets, demonstrating its robustness to propagation perturbations. At a noise level of 0.10, CPNS remains competitive with the best-performing baselines, indicating stable performance under stronger noise. In particular, multi-frequency learning effectively reduces performance degradation under severe perturbations. This is because multi-frequency learning constrains low-frequency consensus and high-frequency residual information separately. Low-frequency signals stabilize shared propagation patterns across perturbed views. High-frequency signals preserve local variations weakened by over-smoothing. Therefore, this mechanism preserves stable and discriminative propagation information under noise. Moreover, multi-frequency learning consistently improves performance across different propagation backbones. These results show that CPNS reduces discriminative information loss during repeated aggregation.

Table 4: On GossipCop, Macro-$F_1$ comparison with baselines under Gaussian noise.

| Methods | Macro-$F_1$ | $\sigma_{0.05}$ | $\sigma_{0.10}$ |
|---|---|---|---|
| Bi-GCN (Bian et al., 2020) | 0.932 | 0.805 | 0.701 |
| UPFD-GAT (Dou et al., 2021) | 0.866 | 0.737 | 0.676 |
| UPFD-SAGE (Dou et al., 2021) | 0.939 | 0.872 | 0.781 |
| UPFD-GCN (Dou et al., 2021) | 0.826 | 0.818 | 0.655 |
| BREAK (Yin et al., 2025) | 0.937 | 0.839 | 0.609 |
| FGDGNN (Guo et al., 2025) | 0.884 | 0.793 | 0.702 |
| KPG (Zhang et al., 2025) | 0.885 | 0.797 | 0.719 |
| MTAS (Le-Pham et al., 2026) | 0.937 | 0.862 | 0.771 |
| H2DFD (Peng et al., 2026) | 0.939 | 0.770 | 0.663 |
| Semantic-GMAE (Liu et al., 2026b) | 0.939 | 0.852 | 0.736 |
| GAAC (Hu et al., 2026) | 0.887 | 0.784 | 0.693 |
| CPNS (ours) | **0.977** | **0.874** | **0.781** |

Table 5: On PolitiFact, Macro-$F_1$ comparison with baselines under Gaussian noise.

| Methods | Macro-$F_1$ | $\sigma_{0.05}$ | $\sigma_{0.10}$ |
|---|---|---|---|
| Bi-GCN (Bian et al., 2020) | 0.836 | 0.752 | 0.706 |
| UPFD-GAT (Dou et al., 2021) | 0.871 | 0.735 | 0.598 |
| UPFD-SAGE (Dou et al., 2021) | 0.859 | 0.830 | 0.772 |
| UPFD-GCN (Dou et al., 2021) | 0.852 | 0.786 | 0.655 |
| BREAK (Yin et al., 2025) | 0.869 | 0.799 | 0.738 |
| FGDGNN (Guo et al., 2025) | 0.870 | 0.764 | 0.653 |
| KPG (Zhang et al., 2025) | 0.869 | 0.836 | 0.713 |
| MTAS (Le-Pham et al., 2026) | 0.870 | 0.803 | 0.724 |
| H2DFD (Peng et al., 2026) | 0.882 | 0.801 | 0.717 |
| Semantic-GMAE (Liu et al., 2026b) | 0.855 | 0.822 | 0.755 |
| GAAC (Hu et al., 2026) | 0.868 | 0.833 | 0.703 |
| CPNS (ours) | **0.909** | **0.848** | **0.768** |

**(3) Comparison with Node Selection Methods**

We compared CPNS-Selector with CSDA (Gong et al., 2026), KPG (Zhang et al., 2025), and HCSL (Liu et al., 2026a). Only the node selection strategy is replaced in each comparison. Note that the selector-level and full-model evaluations use different settings and are not directly compar-

able. CPNS-Selector achieves the highest ACC on both datasets (Table 6). It exceeds the strongest baseline by 3.36 and 3.77 percentage points, respectively. These gains stem from the progressive treatment of propagation evidence. Existing selectors determine node importance before sufficient propagation context is learned. Such decisions can suppress valuable nodes at deeper propagation depths. CPNS-Selector instead introduces deeper nodes progressively and adaptively regulates their contributions. This preserves useful deep-node evidence while reducing unreliable responses. The consistent improvements across both datasets further support this progressive selection strategy.

Table 6: Performance comparison of different node selection strategies.

| Methods | Node Selection ACC | |
|---|---|---|
| | GossipCop | PolitiFact |
| CPNS-Selector | **0.9757** | **0.9434** |
| CSDA-Selector (Gong et al., 2026) | 0.9267 | 0.8929 |
| KPG-Selector (Zhang et al., 2025) | 0.9392 | 0.9020 |
| HCSL-Selector (Liu et al., 2026a) | 0.9421 | 0.9057 |

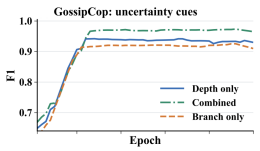


Figure 5: Macro-$F_1$ trajectories under different signals on GossipCop.

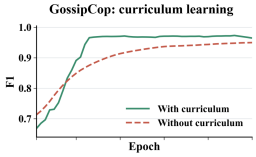


Figure 6: Macro-$F_1$ trajectories with and without curriculum learning on GossipCop.

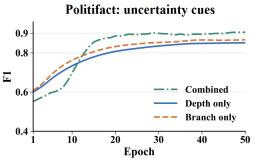


Figure 7: Macro-$F_1$ trajectories under different signals on PolitiFact.

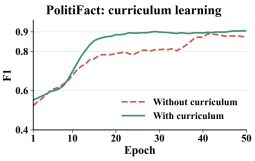


Figure 8: Macro-$F_1$ trajectories with and without curriculum learning on PolitiFact.

### 4.4 Ablation Experiments

**(1) Propagation-aware node selection.**

**w/o Routing.** We compare different routing signals to verify the effect of adaptive routing. Figures 5 and 7 show that combining depth and routing uncertainty achieves the best Macro-$F_1$ on both datasets. Depth provides stable structural information, while routing uncertainty reflects the current learning state. Their combination better controls node propagation, showing the advantage of our adaptive routing strategy.

**w/o Curriculum.** We remove curriculum learning to verify the effect of progressive propagation. Figures 6 and 8 show that curriculum learning improves Macro-$F_1$ and leads to more stable convergence on both datasets. Without curriculum learning, nodes at different depths are introduced at the same time, which may affect the learning of stable propagation patterns. Our curriculum introduces nodes from shallow to deep, leading to more stable and effective propagation learning.

**(2) Candidate node selection.** Candidate node selection reduces LLM evaluations without discarding structurally informative nodes. Figure 9 compares morphology-based sampling with Random and single-cue strategies. Random ignores propagation structure, while single cues describe only one structural property. Our morphology descriptor combines size, depth, branching, and burst-related statistics. These cues jointly characterize each candidate's structural role within the propagation graph. Importantly, depth is considered jointly with other cues rather than independently. Thus, deeper nodes are not directly treated as less valuable candidates. Morphology-based sampling improves Macro-$F_1$ on both datasets while reducing unnecessary LLM evaluations. This supports structural screening before subsequent node evidence evaluation.

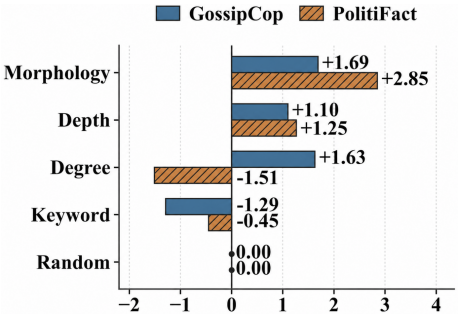


Figure 9: Macro-$F_1$ changes for different node selection strategies relative to Random. Positive values indicate gains, while negative values indicate degradation. Blue solid and orange hatched bars denote GossipCop and PolitiFact, respectively.

Table 7: Comparison of selected nodes under different selection strategies.

| Method | ID | Depth | Node Information | Method | ID | Depth | Node Information |
|---|---|---|---|---|---|---|---|
| CPNS-selection | 66 | 3 | Dominion voting conspiracy and U.S. political controversy. | KPG-selection | 21 | 1 | Highly similar to Node 42; shallow political discussion. |
| CPNS-selection | 50 | 2 | Biden ballots, election fraud, and Michigan election dispute. | KPG-selection | 46 | 1 | Trump and New York business investigation. |
| CSDA-selection | 42 | 1 | False tweets, constitutional interpretation, and political claims. | CSDA-selection | 50 | 2 | Biden ballots, election fraud, and Michigan election dispute. |

### 4.5 GNN Backbone Selection

To identify a suitable GNN backbone, we compare GAT, GCN, and GraphSAGE under identical settings. Figures 10 and 11 report their Macro-$F_1$ trajectories during the first 50 epochs. On GossipCop, GraphSAGE reaches 0.9772, compared with 0.9766 and 0.9691 for GAT and GCN. On PolitiFact, GraphSAGE achieves 0.9091, substantially exceeding GAT at 0.8076 and GCN at 0.8148. GraphSAGE also converges rapidly while maintaining strong performance across both datasets. Its sampled neighborhood aggregation also matches our progressively exposed propagation graphs. Accordingly, we adopt GraphSAGE as the backbone for all subsequent experiments.

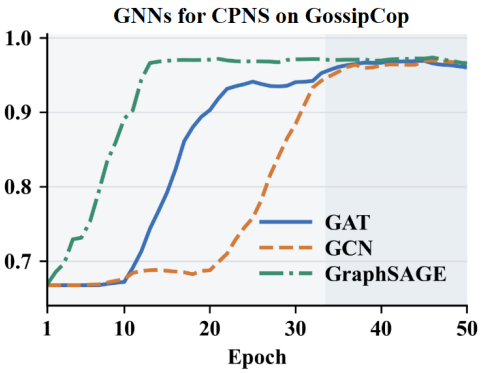


Figure 10: Macro-$F_1$ convergence curves of different GNN backbones on GossipCop.

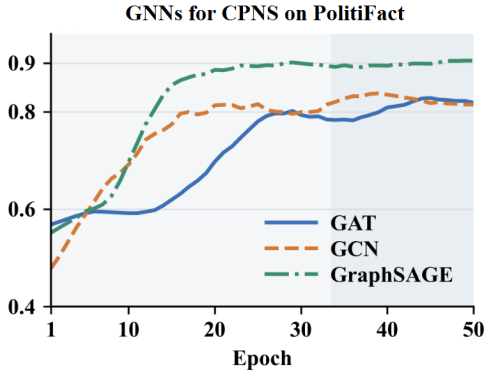


Figure 11: Macro-$F_1$ convergence curves of different GNN backbones on PolitiFact.

### 4.6 Case Study

To verify the effectiveness of CPNS in selecting valuable nodes, we analyze PolitiFact graph #171. As shown in Table 7, CPNS selects nodes 66 and 50 at depths 3 and 2. These nodes cover voting-system controversies, election fraud claims, and regional election disputes. In contrast, KPG selects nodes 21 and 46, both at depth 1. Its selections mainly focus on shallow propagation and local political discussions. CSDA selects nodes 42 and 50 at depths 1 and 2, respectively. Overall, CPNS captures deeper propagation patterns while preserving diverse event-related nodes. It effectively avoids ignoring informative nodes in deeper propagation layers. This broader propagation learning further alleviates feature over-smoothing in propagation graphs. Further analysis is provided in Appendix C.

## 5 Conclusion

In this paper, we propose CPNS to select valuable nodes on noisy propagation graphs for fake news detection. CPNS progressively selects valuable nodes across propagation depths through curriculum learning, while utilizing evidence-aware weighting to suppress noisy nodes. Multi-frequency consistency preserves the complete and valuable information in propagation representations. Experimental results show that our method improves Macro-$F_1$ by 1.54% and 2.15% over strong existing baselines on GossipCop and PolitiFact, respectively.

## AI Use Statement

We used generative AI tools to polish the writing and improve the clarity of expression. All AI-assisted revisions were reviewed and verified by the authors, who take full responsibility for the final content of this work.

## Reproducibility Statement

Section 3 describes the overall framework of CPNS. Appendix B provides the experimental settings, evaluation metrics, and implementation details. The source code is provided in the supplementary material to facilitate reproducibility.

## References

Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 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, 2020. doi: 10.1609/aaai.v34i01.5393.

Junyang Chen, Yueqian Li, Ka Chung Ng, Huan Wang, and Liang-Jie Zhang. Toward multimodal fake news detection by multi-perspective rationale generation and verification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 66–74, 2026. doi: 10.1609/aaai.v40i1.36965.

Chaoqun Cui and Caiyan Jia. 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, 2024. doi: 10.1609/aaai.v38i1.27757.

Yingtong Dou, Kai Shu, Congying Xia, Philip S. Yu, and Lichao Sun. User preference-aware fake news detection. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2051–2055, 2021. doi: 10.1145/3404835.3462990.

Shuzhi Gong, Richard O. Sinnott, Jianzhong Qi, and Cecile Paris. Invariant subgraphs for cross-domain fake news detection via causal disentanglement. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 20, pages 910–922, 2026. doi: 10.1609/icwsm.v20i1.42673.

Mei Guo, Chen Chen, Chunyan Hou, Yike Wu, and Xiaojie Yuan. FGDGNN: Fine-grained dynamic graph neural network for rumor detection on social media. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5676–5687. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.296.

Jie Hu, Langsha Zhu, Mei Yang, Bingbing Tang, Shicheng Dai, and Chenglong Zhang. GAAC: A robust misinformation detection framework via graph attention and adaptive contrastive learning. *Knowledge-Based Systems*, art. 115664, 2026. doi: 10.1016/j.knosys.2026.115664.

Hoang-Trung Le-Pham, The-Phong Nguyen, Anh-Duy Tran, and Kim-Hung Le. MTAS: A temporal-aware multiview graph attention framework for early rumor detection on social media. *Information Fusion*, 126:103598, 2026. doi: 10.1016/j.inffus.2025.103598.

Ping Liu, Hui Song, Shufeng Hao, Xiaoning Hao, Zexu Zhang, and Usman Naseem. HCSL: Rumor detection by integrating intra-sample curriculum learning and hierarchical semantic learning. In *Proceedings of the ACM on Web Conference*, pages 9068–9078, 2026a. doi: 10.1145/3774904.3793016.

Qiang Liu, Xiang Tao, Liang Wang, Shu Wu, and Liang Wang. Robust and generalizable rumor detection with semantic evolving graph masked autoencoder. *Pattern Recognition*, art. 112995, 2026b. doi: 10.1016/j.patcog.2025.112995.

Mengyao Peng, Yiping Ma, Hongyan Gu, Xinyi Zhang, Yedi Ma, and Xinli Huang. H2DFD: Self-supervised fake news detection via a novel hyperbolic hypergraph diffusion model. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 2026. doi: 10.1109/ICASSP55912.2026.11464061.

Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. FakeNewsNet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. *Big Data*, 8(3):171–188, 2020. doi: 10.1089/big.2020.0062.

Lingwei Wei, Dou Hu, Wei Zhou, Zhaojuan Yue, and Songlin Hu. Towards propagation uncertainty: Edge-enhanced bayesian graph convolutional networks for rumor detection. In *Proceedings of the Joint Conference of the Annual Meeting of the ACL and the International Joint Conference on Natural Language Processing*, pages 3845–3854, 2021. doi: 10.18653/v1/2021.acl-long.297.

Jiaying Wu and Bryan Hooi. DECOR: Degree-corrected social graph refinement for fake news detection. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2582–2593, 2023. doi: 10.1145/3580305.3599298.

Ruichao Yang, Xiting Wang, Yiqiao Jin, Chaozhuo Li, Jianxun Lian, and Xing Xie. Reinforcement subgraph reasoning for fake news detection. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2253–2262, 2022. doi: 10.1145/3534678.3539277.

Xiaoyu Yang, Yuefei Lyu, Tian Tian, Yifei Liu, Yudong Liu, and Xi Zhang. Rumor detection on social media with graph structured adversarial learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1417–1423, 2020. doi: 10.24963/ijcai.2020/197.

Junwei Yin, Min Gao, Kai Shu, Wentao Li, Yinqiu Huang, and Zongwei Wang. Graph with sequence: Broad-range semantic modeling for fake news detection. In *Proceedings of the ACM on Web Conference*, pages 2838–2849, 2025. doi: 10.1145/3696410.3714906.

Shu Yin, Peican Zhu, Lianwei Wu, Chao Gao, and Zhen Wang. GAMC: An unsupervised method for fake news detection using graph autoencoder with masking. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 347–355, 2024. doi: 10.1609/aaai.v38i1.27788.

Shu Yin, Yuchen Wang, Dongpeng Hou, Wenxin An, Chao Gao, Xianghua Li, and Zhen Wang. LLM-assisted fake news detection with adaptive boosting framework incorporating contrastive learning. *Information Processing & Management*, art. 104789, 2026. doi: 10.1016/j.ipm.2026.104789.

Litian Zhang, Xiaoming Zhang, Chaozhuo Li, Ziyi Zhou, Jiacheng Liu, Feiran Huang, and Xi Zhang. Mitigating social hazards: Early detection of fake news via diffusion-guided propagation path generation. In *Proceedings of the ACM International Conference on Multimedia*, pages 2842–2851, 2024. doi: 10.1145/3664647.3681087.

Yusong Zhang, Kun Xie, Xingyi Zhang, Xiangyu Dong, and Sibo Wang. Rumor detection on social media with reinforcement learning-based key propagation graph generator. In *Proceedings of the ACM on Web Conference*, pages 2742–2753, 2025. doi: 10.1145/3696410.3714651.

Junyou Zhu, Chao Gao, Ze Yin, Xianghua Li, and Juergen Kurths. Propagation structure-aware graph transformer for robust and interpretable fake news detection. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4652–4663, 2024. doi: 10.1145/3637528.3672024.

Junyou Zhu, Chao Gao, Ze Yin, Xianghua Li, Zhen Wang, and Jürgen Kurths. Noise-filtering enhanced graph transformer for robust fake news detection. *IEEE Transactions on Knowledge and Data Engineering*, 2026. doi: 10.1109/TKDE.2026.3677544.

# A LLM-based Node Evidence Classification

**Node Value Classification Using an LLM.** We summarize propagation morphology by

$$\mathbf{m}_i = [m_{i,1}, \ldots, m_{i,10}]^\top = \phi_{\text{morph}}(G_i) \in \mathbf{R}^{10}, \tag{16}$$

where $m_{i,1:10}$ encode, respectively, node count, maximum depth, depth-distribution entropy, maximum out-degree, hub depth, normalized maximum out-degree, first-level branch imbalance, maximum subtree fraction, deep-node ratio, and deep-burst score. Let $b_{i,v}$, $\eta_{i,v}$, and $\kappa_{i,v}$ denote local burstiness, subtree size, and the fact-checking-keyword indicator, respectively. The morphology-conditioned node score is

$$s_{i,v} = \phi_{\text{sel}}(d_i(v), b_{i,v}, \eta_{i,v}, \kappa_{i,v}; \mathbf{m}_i)\,, \tag{17}$$

where $\phi_{\text{sel}}$ is the selector scoring rule. The score is evaluated over all nodes, while only the fixed candidate budget is sent to the LLM; the assessment is label-isolated as described in Appendix A.

The candidate set is the top-$k$ nodes under this score:

$$\begin{aligned} \mathcal{C}_i &= \{v \in V_i : \text{rank}_{V_i}(s_{i,v}) \le k\}\,, \\ c_{i,v} &= \mathbf{1}[v \in \mathcal{C}_i]\,. \end{aligned} \tag{18}$$

Here $\text{rank}_{V_i}$ is the descending rank over $V_i$ (ties follow the fixed selector order), $k$ is the candidate budget, and $\mathbf{1}[\cdot]$ is the indicator. Scoring still scans $V_i$, whereas only $O(k)$ language-model assessments are performed.

For each $v \in \mathcal{C}_i$, LLM receives the root text, morphology summary, candidate text, and local parent–child context, without $y_i$ or a graph-level Real/Fake request. It returns $j_{i,v} \in \{\text{evidence}, \text{noise}, \text{ambiguous}\}$, $\text{val}_{i,v}, \text{rel}_{i,v} \in [0, 1]$, and a rationale.

**Model and execution.** We employ DeepSeek-4V-7B as an offline evidence assessor with 7B parameters. The LLM only classifies nodes based on their numerical representations. It does not generate opinions or additional content. The prompt is defined as follows:

```
You are a node-level evidence assessor for fake-news propagation
graphs.
Your task is to classify each node as useful evidence, noise, or
ambiguous information for downstream fake-news detection.
Never use or infer any hidden ground-truth label. Output valid JSON
only.
```

Table 8: Statistics of the GossipCop and PolitiFact datasets.

| Statistical metric | GossipCop | PolitiFact |
|---|---|---|
| Number of propagation graphs | 5,464 | 314 |
| Real/Fake | 2,732/2,732 | 157/157 |
| Number of nodes | 314,262 | 41,054 |
| Average number of nodes (range) | 57.52 (3–199) | 130.75 (3–497) |
| Training/Validation/Test | 1,092/546/3,826 | 62/31/221 |
| Deep propagation graphs ($d \ge 3$) | 1,638 (29.98%) | 228 (72.61%) |

# B Experimental Setting

## B.1 Dataset Statistics

The statistics of the GossipCop and PolitiFact datasets are reported in Table 8.

## B.2 Parameter Settings

Each experiment was run independently with seeds 42–46, and results were averaged over the fixed data splits. We trained for at most 100 epochs using a batch size of 128, a hidden dimension of

Table 9: Architecture, optimization, and method-specific settings.

| Category | Parameter | Setting |
|---|---|---|
| **Architecture** | | |
| | GNN backbone | GraphSAGE |
| | Input feature dimension | 4096 |
| | Hidden dimension | 256 |
| | Message-passing layers | 2 |
| | Propagation experts per layer ($K$) | 3 |
| | Gumbel–Softmax temperature ($\tau$) | 1.0 |
| | Dropout rate | 0.5 |
| | Normalization | Layer normalization |
| | Residual connection | Disabled |
| **Optimization** | | |
| | Optimizer | Adam |
| | Learning rate | $1 \times 10^{-3}$ |
| | Weight decay | $1 \times 10^{-2}$ |
| | Batch size | 128 |
| | Maximum epochs | 100 |
| | Checkpoint selection | Highest validation ACC plus validation $F_1$-score |
| **Curriculum learning** | | |
| | Stage-1 depth threshold ($Q_1$) | 1 |
| | Stage-2 depth threshold ($Q_2$) | 2 |
| | Routing uncertainty weight ($\beta$) | 1.0 |
| **Routing regularization** | | |
| | Regularization coefficient | $\lambda_{\text{reg}} = 0.02$ |
| **Morphology descriptor** | | |
| | Scale-aware features | Enabled |
| | Scale feature dimension | 2 |
| | Scale variables | $\log(1 + \lvert V \rvert)$ and maximum propagation depth |
| **Multi-frequency consistency** | | |
| | Frequency contrastive loss weight | $\lambda_{\text{freq}} = 0.05$ |
| | InfoNCE temperature | $\tau_{\text{freq}} = 0.2$ |

256, two GNN layers, dropout 0.5, and weight decay 0.01. GraphSAGE was used as the propagation backbone. Full architecture and training settings are listed in Table 9.

### B.3 Evaluation Metrics and Robustness Analysis

**Evaluation Metrics.** We evaluate the performance of our method using four metrics: ACC, Macro-$F_1$, Precision, and Recall. For multi-class classification, let $C$ denote the number of classes, and let $TP_c$, $TN_c$, $FP_c$, and $FN_c$ represent the numbers of true positives, true negatives, false positives, and false negatives for class $c$, respectively.

ACC measures the proportion of correctly classified samples among all samples and is defined as

$$\text{ACC} = \frac{\sum_{c=1}^{C} TP_c}{N}, \tag{19}$$

where $N$ denotes the total number of samples.

Precision measures the proportion of correctly predicted positive samples among all samples predicted as positive. Recall measures the proportion of actual positive samples that are correctly identified. We report their macro-averaged values, defined as

$$P_c = \frac{TP_c}{TP_c + FP_c}, \qquad R_c = \frac{TP_c}{TP_c + FN_c}, \tag{20}$$

$$\text{Precision} = \frac{1}{C}\sum_{c=1}^{C} P_c, \qquad \text{Recall} = \frac{1}{C}\sum_{c=1}^{C} R_c. \tag{21}$$

Macro-$F_1$ computes the $F_1$-score independently for each class and then averages the resulting scores, assigning equal importance to all classes:

$$F_{1,c} = \frac{2P_cR_c}{P_c + R_c}, \tag{22}$$

$$\text{Macro-}F_1 = \frac{1}{C}\sum_{c=1}^{C} F_{1,c}. \tag{23}$$

For all four metrics, higher values indicate better classification performance.

**Robustness to Gaussian Noise.** To evaluate the robustness of our method against input perturbations, we conduct additional experiments by introducing additive Gaussian noise at two different noise levels. Specifically, the perturbed input is formulated as

$$\widetilde{\mathbf{x}} = \mathbf{x} + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}), \tag{24}$$

where $\mathbf{x}$ denotes the original input, $\widetilde{\mathbf{x}}$ represents the perturbed input, and $\boldsymbol{\epsilon}$ denotes Gaussian noise. We consider two noise levels, with standard deviations $\sigma \in \{0.05, 0.10\}$, to assess the robustness of the model under different degrees of perturbation. The classification performance is evaluated using the four metrics defined above.

### B.4 Baseline Details

We compared CPNS with two categories of representative graph-structured fake news detection methods.

**(1) Propagation representation methods.** These methods learn discriminative representations from news propagation structures.

**Bi-GCN** (Bian et al., 2020) models rumor propagation from two complementary directions. It constructs top-down and bottom-up graphs from observed propagation trees. Two graph convolutional networks separately capture propagation and dispersion patterns. Source-post information is incorporated to strengthen node representations. The bidirectional representations are finally fused for rumor classification.

**UPFD** (Dou et al., 2021) incorporates user preferences into propagation-based fake news detection. User preferences are derived from users' historical social engagements. These preferences reflect users' tendencies to interact with particular news content. News content and user preferences are jointly modeled over propagation graphs. Their combination provides complementary semantic and behavioral information for detection.

**CGAT** (Yang et al., 2020) improves robustness against malicious manipulation of social structures. It constructs heterogeneous graphs containing users, posts, comments, and their interactions. An adversarial graph generator produces constrained structural perturbations during training. These perturbations simulate realistic camouflage behaviors in rumor propagation. Adversarial training encourages robust representations against structural manipulation.

**EBGCN** (Wei et al., 2021) explicitly considers uncertainty in observed propagation relations. It uses Bayesian inference to estimate latent reliability for propagation edges. Inferred edge weights control information aggregation during graph convolution. Unreliable relations therefore contribute less information during message passing. Edge-wise consistency training further regularizes the inferred propagation relations.

**FGDGNN** (Guo et al., 2025) represents rumor propagation using sequential dynamic graph snapshots. It captures fine-grained temporal information within individual propagation snapshots. Edge-aware graph encoding models temporal differences among propagation relations. Representation transformation captures evolving information across consecutive graph snapshots. Thus, both intra-graph and inter-graph propagation dynamics are jointly modeled.

**MTAS** (Le-Pham et al., 2026) jointly models semantic, structural, and temporal propagation information. It extracts complementary representations from heterogeneous propagation structures. A textual branch preserves semantic information from news-related content. Temporal features characterize propagation dynamics through observed time intervals. These representations are integrated through attention-based fusion for early detection.

**H2DFD** (Peng et al., 2026) models higher-order interactions using hypergraph-based propagation structures. Hypergraphs capture interactions beyond conventional pairwise graph connections. Hyperbolic representations characterize hierarchical patterns within information propagation. Diffusion-based learning propagates information across higher-order structural relations. Self-supervised objectives further improve propagation representation learning.

**GAAC** (Hu et al., 2026) combines graph attention, adaptive augmentation, and contrastive learning. Graph attention extracts informative representations from propagation structures. Adaptive augmentation generates diverse views while preserving important structural information. Adversarial hard samples provide challenging examples for contrastive representation learning. These mechanisms improve robustness against structural noise and perturbations.

**(2) Structure denoising and key-information methods.** These methods suppress noisy information while preserving informative structures and semantic patterns.

**BREAK** (Yin et al., 2025) constructs broad-range semantic graphs from news sentences. Its structure denoising module suppresses irrelevant semantic relations between sentences. Feature denoising removes redundant information from learned semantic representations. Graph and sequence representations jointly guide the feature denoising process. Both modules produce cleaner semantic representations for fake news detection.

**KPG** (Zhang et al., 2025) learns key propagation graphs for fake news detection. It addresses sparse early propagation and redundant mature propagation structures. A Candidate Response Generator produces informative candidate propagation patterns. An Ending Node Selector identifies influential locations for generated responses. Reinforcement learning guides graph generation using detection performance as feedback.

**SEE-GraphMAE** (Liu et al., 2026b) models semantic evolution during rumor propagation. It masks propagation information and reconstructs it through self-supervised learning. Reconstruction captures both local semantic changes and global evolution patterns. Learned semantic features are combined with structural propagation representations. Test-time training further improves generalization to emerging propagation patterns.

**CSDA** (Gong et al., 2026) targets domain shifts in cross-domain fake news detection. It seeks invariant propagation substructures shared across different domains. A graph-based mask generator identifies informative nodes and propagation edges. Domain-specific and spurious structural information is suppressed during representation learning. Contrastive learning further supports adaptation with limited target-domain samples.

**HCSL** (Liu et al., 2026a) combines intra-sample curriculum learning with hierarchical semantic learning. Curriculum learning progressively emphasizes informative components within each propagation sample. Hierarchical learning captures semantic evidence at multiple representation levels. Less informative components receive reduced influence during representation learning. Their combination yields more discriminative representations for rumor detection.

### B.5 Multi-Frequency Consistency under Feature Noise

With multi-frequency consistency (MFC), CPNS's degradation on GossipCop at $\sigma = 0.05$, 0.10, and 0.20 decreases from 9.91%, 29.18%, and 42.76% to 6.86%, 21.68%, and 35.62%, respectively; on PolitiFact at $\sigma = 0.10$ and 0.20, it decreases from 20.09% and 45.85% to 5.22% and 17.68%, respectively. Gains vary across external models, so the effect depends on encoder structure and data distribution. Tables 4 and 5 report the complete per-backbone robustness results, making the cross-model pattern explicit.

Table 10: Macro-$F_1$ of each model on GossipCop and the gain $\Delta$ after adding the multi-frequency consistency method.

| Model | Macro-$F_1$ | $\Delta_{0.05}$ (%) | $\Delta_{0.10}$ (%) | $\Delta_{0.20}$ (%) |
|---|---|---|---|---|
| Bi-GCN (Bian et al., 2020) | 0.9315 | +3.68 | +6.00 | +4.78 |
| UPFD-GAT (Dou et al., 2021) | 0.8657 | +9.05 | +8.01 | +3.41 |
| UPFD-SAGE (Dou et al., 2021) | 0.9386 | +1.57 | +3.46 | +2.82 |
| UPFD-GCN (Dou et al., 2021) | 0.8262 | +0.22 | +0.72 | +0.06 |
| BREAK (Yin et al., 2025) | 0.9373 | +4.61 | **+16.12** | +13.05 |
| FGDGNN (Guo et al., 2025) | 0.8838 | +0.44 | +0.95 | +4.87 |
| KPG (Zhang et al., 2025) | 0.8854 | +0.19 | +0.88 | +5.19 |
| MTAS (Le-Pham et al., 2026) | 0.9370 | +1.27 | +11.49 | **+18.30** |
| H2DFD (Peng et al., 2026) | 0.9388 | **+11.19** | +11.92 | +8.10 |
| Semantic-GMAE (Liu et al., 2026b) | 0.9391 | +4.25 | +3.05 | +1.42 |
| GAAC (Hu et al., 2026) | 0.8873 | +1.18 | +1.73 | +1.66 |
| CPNS (ours) | **0.9772** | +2.9 | +7.4 | +7.0 |

Table 11: Macro-$F_1$ of each model on PolitiFact and the gain $\Delta$ after adding the multi-frequency consistency method.

| Model | Macro-$F_1$ | $\Delta_{0.05}$ (%) | $\Delta_{0.10}$ (%) | $\Delta_{0.20}$ (%) |
|---|---|---|---|---|
| Bi-GCN (Bian et al., 2020) | 0.8360 | +1.71 | +1.69 | +1.54 |
| UPFD-GAT (Dou et al., 2021) | 0.8710 | 0.00 | +1.71 | +0.25 |
| UPFD-SAGE (Dou et al., 2021) | 0.8591 | +4.86 | +1.69 | +1.38 |
| UPFD-GCN (Dou et al., 2021) | 0.8517 | +1.71 | 0.00 | +1.76 |
| BREAK (Yin et al., 2025) | 0.8687 | +1.49 | +5.50 | +6.48 |
| FGDGNN (Guo et al., 2025) | 0.8704 | +3.38 | +0.09 | +10.56 |
| KPG (Zhang et al., 2025) | 0.8691 | +3.38 | +0.09 | +10.56 |
| MTAS (Le-Pham et al., 2026) | 0.8701 | +1.95 | +7.04 | +7.97 |
| H2DFD (Peng et al., 2026) | 0.8816 | +1.59 | +1.67 | +8.44 |
| Semantic-GMAE (Liu et al., 2026b) | 0.8547 | +4.71 | +9.61 | +1.54 |
| GAAC (Hu et al., 2026) | 0.8688 | +3.38 | +0.09 | +10.56 |
| CPNS (ours) | **0.9091** | **+7.5** | **+10.2** | **+23.5** |

## C Case Study of the Dataset

These cases in Table 12 show a consistent pattern. First, deep nodes still contain non-negligible valuable information, contributing 33.3%, 28.0%, and 21.4%, respectively. Although these contributions are close to the proportions of deep nodes and do not indicate higher importance per node, they show that valuable information is not limited to shallow regions. The raw texts further show clear differences across propagation depths. Shallow nodes mainly contain source-level content, such as news or show descriptions, while deep nodes contain more user reactions, reposts, and discussions. Therefore, deep regions provide different and valuable information that should not be ignored. Simply limiting the propagation depth may lose such information, while increasing the propagation depth may reduce the discriminative ability of node representations.

Table 12: Case Study on Node Analysis.

| Case | Nodes | Deep nodes | Deep contribution | Full cosine | Subgraph cosine | $\Delta$ cosine |
|---|---|---|---|---|---|---|
| GossipCop #208 | 31 | 10/31 | 33.3% | 0.964 | 0.986 | +0.022 |
| GossipCop #13 | 24 | 7/24 | 28.0% | 0.957 | 0.965 | +0.008 |
| PolitiFact #2 | 312 | 68/312 | 21.4% | 0.926 | 0.931 | +0.006 |