arXiv:2602.13279v2 [cs.SI] 22 Mar 2026

通过虚拟节点诱导的边预测实现大语言模型增强的谣言检测

Jiran Tao 数据科学与人工智能系 香港理工大学 中国香港九龙红磡22117758r@connect.polyu.hk
Cheng Wang 数学科学学院 上海交通大学 中国上海 chengwang@sjtu.edu.cnBinyan Jiang 数据科学与人工智能系 香港理工大学 中国香港九龙红磡 by.jiang@polyu.edu.hk

摘要

谣言在社交网络上的迅速传播对信息完整性构成了重大威胁。尽管谣言传播会形成复杂的结构模式，现有检测方法往往无法捕捉文本连贯性与传播动态之间错综复杂的相互作用。当前方法通常通过孤立的文本嵌入来表示节点，忽视了整个传播路径中的语义流动。为弥补这一不足，我们提出一种新颖的框架，将大语言模型（LLM）作为结构增强层，应用于基于图的谣言检测。不同于传统方法，我们的框架利用大语言模型评估信息子链，并策略性地在图中引入一个虚拟节点。这一结构调整将潜在语义模式转化为显式的拓扑特征，从而有效捕捉文本连贯性，而这一直以来都是图神经网络（GNN）难以获取的信息。为确保可靠性，我们设计了一种结构化提示框架，在保持图学习性能稳健的同时，减轻大语言模型固有的偏见。此外，我们提出的框架与模型无关，不受任何特定图学习算法或大语言模型的限制。其即插即用的特性使其能够轻松集成未来进一步微调的大语言模型和图技术，从而有望在无需修改原始算法的情况下提升预测性能。

1 引言

社交媒体的迅速扩张彻底改变了信息共享方式，但与此同时也加快了谣言传播，威胁着信息可信度和社会稳定（Ziari 和 Charkari，2025年）。在社交网络中检测谣言是一项至关重要但颇具挑战性的任务，因为谣言

通常会随时间沿树状图结构向下传播，其中节点（代表用户或帖子）和边（表示回复或转发等互动）反映了信息流动。这种时间特性与文本数据的语义复杂性相结合，因此需要采用能够有效捕捉结构模式、时间演化和深层语义内容的建模方法。

传统谣言检测方法虽取得了一定成效，但仍受到显著局限的制约。依赖人工设计特征的早期机器学习方法难以适应社交媒体数据固有的多样性和噪声（Zubiaga等人，2018年）。同样，传统图神经网络主要用于建模局部图结构，往往难以捕捉不同长度子链中体现的细微谣言传播路径和扩散模式。大语言模型擅长从文本中提取丰富的语义特征，为弥补传统图神经网络在语义方面的不足带来了良好机遇（Wang等人，2024年）。然而，我们的实验揭示了大语言模型单独用于谣言检测时的一项关键局限：由于预训练数据和强化学习目标不同，各模型表现出的倾向性存在显著差异。有些大语言模型十分保守、规避风险，另一些则更为激进或宽松。因此，单独采用基于大语言模型的方法进行谣言检测，本质上可能缺乏可靠性；附录G展示了一个失败案例。鉴于这些相互矛盾的行为以及网络数据本身的复杂性，如何有效利用大语言模型提升谣言检测性能，仍是一个关键且尚待解决的问题。为应对这些挑战，我们提出了一种新颖的模型无关框架，将大语言模型与图神经网络相结合，利用结构化的子链传播模式提升谣言检测准确率。

我们的方法利用大语言模型分析推文子链中的信息流，在利用其丰富内部知识的同时，捕捉关键传播模式。我们采用结构化提示框架来促进网络数据处理，并减轻大语言模型的分类偏差；不过，我们的主要创新在于对谣言图进行架构增强。我们引入一个表示谣言状态的虚拟节点，并利用大语言模型为每个子链生成的细粒度谣言概率，扩充现有图结构。该方法效率很高，因为它仅依赖大语言模型的 API 调用，无需本地模型调优或高强度存储基础设施。随后，我们对增强后的图进行处理，以获得稳健的节点表示，从而通过根节点与虚拟节点之间的链接预测，实现准确的谣言检测。总之，我们的研究有两项主要贡献：（1）我们提出了一种开创性的模型无关框架，协同整合大语言模型与图神经网络用于谣言检测，从而有效融合语义信息与结构信息。（2）我们提出了一种创新方法，通过使用虚拟节点和基于子链的连接重构图，捕捉尚未充分利用的传播特征，从而增强对复杂谣言扩散模式的检测能力。

2 相关工作

2.1 谣言检测

谣言检测从早期基于特征的机器学习方法发展到了深度学习方法。最初的技术使用人工设计的特征和支持向量机等模型，但难以应对社交媒体的复杂性。深度学习引入了用于提取空间特征的卷积神经网络，以及用于时间建模的循环神经网络/长短期记忆网络，这一方向由Ma等人（2016）开创。BERT等Transformer模型提升了语义理解能力，而Ma等人（2017）则通过核学习纳入了传播结构。

2.2 大语言模型在虚假新闻检测中的应用

大语言模型在假新闻检测中展现出显著能力，推动相关技术超越传统方法。一种方法将大语言模型用作评判者，利用其深层语言理解能力大规模评估信息（Zhou等人，2023；Li等人，2024）。另一研究方向通过参数高效微调提升大语言模型的性能，例如Cheung和Lam（2023）的FactLLAMA纳入了外部知识，而Tian等人（2025）则通过选择性筛选解决数据质量问题。除了评估之外，大语言模型还能通过数据增强提升检测系统（Lai等人，2024），并催生了新的框架，例如Ma等人（2024b）提出的方法，通过对竞争视角进行建模来实现核实。

2.3 谣言检测中的图方法

基于图的方法对于社交媒体上的谣言检测至关重要，能够通过网络结构捕捉信息传播模式。谣言通常具有独特的结构特征，例如转发链或分支回复线程，这些特征天然适合用图模型表示。图卷积网络（GCN）聚合相连节点的特征，以生成鲁棒的嵌入。例如，Bian等人（2020）提出了双向图卷积网络，用于捕捉双向信息流，从而提升关键节点识别能力。类似地，Wu等人（2020）引入了带有注意力机制的传播图神经网络，以建模复杂的非序列化谣言扩散。Sun等人（2022）提出了图对抗对比学习框架，通过对抗特征变换提升鲁棒性。近期，Liu等人（2024）设计了一种采用二部图的图神经网络，用于建模用户相关性及树状传播模式，并将社交上下文与扩散拓扑相结合。Ma等人（2024a）提出了图采样与聚合模型（GSMA），通过引入动态注意力、位置编码和情感感知特征，增强GraphSAGE（Hamilton等人，2017），以提升谣言检测效果；而SePro Zeng等人（2025）则将大语言模型推理与GAT式聚合相结合，以整合丰富的社交语境。

3 问题陈述

令 $\mathcal{N} = \{N_1, N_2, \ldots, N_s\}$ 表示源新闻文章的集合，其中每个 $N_i \in \mathcal{N}$ 代表一条新闻条目。每个 N_i 都关联着一组反应帖 $\mathcal{R}_i = \{R_{i1}, R_{i2}, \ldots, R_{it_i}\}$，在每个原始新闻帖下形成多层次讨论结构。具体而言：1）每个新闻帖包含大量用户回复；2）这些回复既可能直接回应新闻帖，也可能嵌套在现有回复下，形成子回复，从而构成树状对话结构。这种树状传播结构与谣言的传播方式完全一致。如图 1(a)所示，顶部的黄色根节点代表源新闻，下方的子节点均为反应帖。

在谣言检测任务中，该问题被表述为一个二分类任务。目标是使用带有真实标签 $y \in \{$谣言，非谣言$\}$的新闻实例训练模型，使其能够准确预测未见测试新闻条目的标签。我们的工作致力于开发一种模型无关的谣言检测框架，以便与各种图学习算法和大语言模型无缝集成。

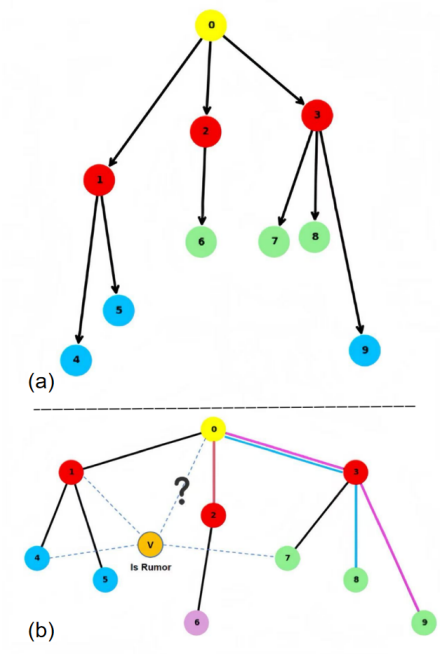


图 1：社交媒体中的谣言传播模式。

4 算法

对于每条源新闻，构建一个有向图 $\mathbb{G} = (\mathbb{V}, \mathbb{E})$：$\mathbb{V} = \{v_0, v_1, \ldots, v_n\}$ 是节点集合，其中 v_0 是源新闻（根节点），每个子节点 $v_i \in \mathbb{V} \backslash \{v_0\}$ 表示一个回复帖。$\mathbb{E}$ 是边集合，边的方向与信息传播方向一致。与节点 v_i 关联的文本记为 text(v_i)。每个节点 v_i 都被赋予一个特征向量 x_i，该向量由 BERT 从最后一层的 [CLS] 表示中提取，如下所示：

$$x_i = \text{BERT}(\text{text}(v_i)).$$

子链构造： 对于每个子节点 $v_i \in \mathbb{V} \backslash \{v_0\}$，都存在一条从根节点 v_0 到 v_i的唯一路径（子链），表示为：

$$\text{path}(v_0, v_i) = (v_0, v_{p_1}, v_{p_2}, \ldots, v_{p_k}, v_i),$$

其中，v_{p_j} 是路径上的中间节点，而 $k \geq 0$。子链的文本信息使用分隔符 [SEP]连接：

$$\begin{aligned}\text{info}(\text{path}(v_0, v_i)) = \text{text}(v_0)\ [\text{SEP}]\ \text{text}(v_{p_1}) \cdots \\ \cdots\ [\text{SEP}]\ \text{text}(v_i).\end{aligned}$$

这种链状信息流捕捉了从根节点到每个子节点的丰富上下文信息。利用这种结构，我们的方法旨在增强大语言模型和图学习算法有效检测谣言的能力。如图 1(b) 所示，从根节点到子节点 2 的红色路径表示与子节点 2 关联的子链。类比可知，整个图包含的子链数量与子节点数量相同，每条子链都对应一条从根节点到某个子节点的独特路径。

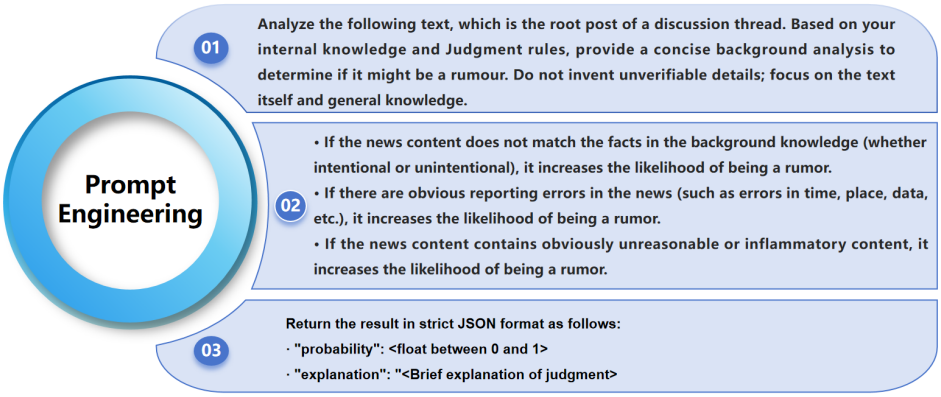


图 2：提示设计方法概览。

图结构中的虚拟节点与边增强：为利用大语言模型的能力和信息流的图结构，我们引入了一个标记为“是谣言”的虚拟节点 v^*，该节点最初没有特征信息。 v^* 的初始特征设为零向量，且其邻居节点不会从 v^*聚合消息。这意味着，即使在双向图神经网络中，虚拟节点也只有指向它的边。然后，我们使用大语言模型处理每条子链，并输出源新闻是谣言的概率。对于每个子节点 v_i，大语言模型会将子链中的文本信息拼接起来，并输出一个概率：

$$P_{\text{rumor}}(v_i) = \text{LLM}(\text{info}(\text{path}(v_0, v_i))), \tag{1}$$

其中， $P_{\text{rumor}}(v_i) \in [0,1]$ 表示基于从根节点到节点 v_i的子链，源新闻是谣言的可能性。对于每个子节点 v_i，如果大语言模型给出的概率超过预设阈值 θ:

$$P_{\text{rumor}}(v_i) > \theta,$$

在 v_i 和 v^*之间建立一条有向边，记为 $(v_i, v^*) \in \mathbb{E}'$，其中 $\mathbb{E}'$ 是添加到图中的新边集合。更新后的图记为 $\mathbb{G}' = (\mathbb{V} \cup \{v^*\}\ \mathbb{E} \cup \mathbb{E}')$。如图1(b)所示，假设大语言模型为子节点1、4和7对应的子链所分配的概率均超过阈值，则在这些节点与虚拟节点之间建立边。通过这些操作，我们引入一个虚拟节点来增强图结构，并利用大语言模型的推理能力，将其与跨子链的谣言传播联系起来。随后，谣言检测问题便转化为预测根节点 v_0 与虚拟节点 v^*之间是否存在连接。接下来，我们将进一步介绍如何从大语言模型中获取概率(1)。已有研究表明，精心设计的提示工程能够增强大语言模型识别谣言的能力（Yan 等，2024；Shehata，2024）。本文提出了一种专门的提示工程框架。如图2步骤1所示，输入提示引导大语言模型利用其内部知识和预定义规则，简要分析根帖的背景，并评估其成为谣言的可能性。为了解决大语言模型对同一新闻采用不同评估标准的问题（Huang，2025；Mohanty，2025），我们提示模型在处理根节点的原始新闻帖时，查询并整理有关该事件的详细背景信息，并将其存储在持久化知识库中。在图2步骤2中，后续的子链评估严格遵循该知识库以及提示中的标准，确保整个过程中的评估一致且可靠。如第5.3节所示，这种提示策略成功提升了大语言模型进行谣言检测的可靠性。图3

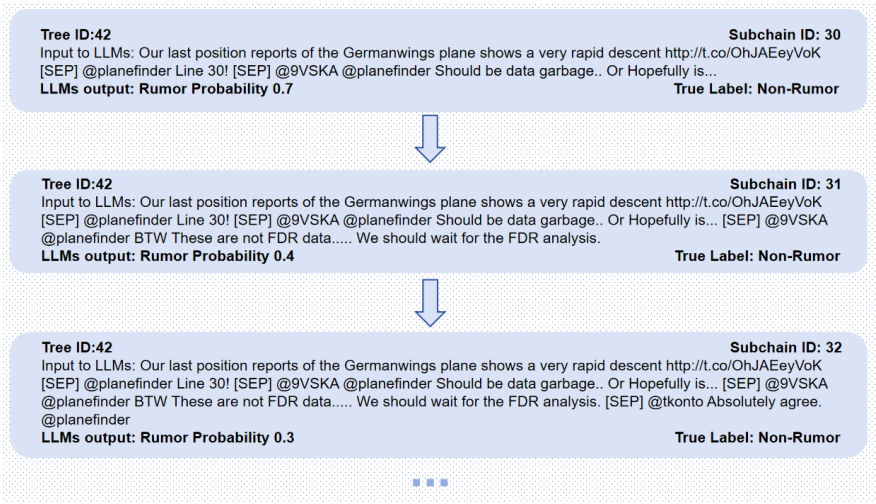


图3：大语言模型处理信息流。

展示了一个示例，说明大语言模型如何利用空中客车A320德国之翼航空公司空难的突发新闻报道，在传播树中评估各子链的谣言概率。根新闻的真实标签为“非谣言”。ID为30、31和32的子链仅在一个子节点上有所不同。可以观察到，随着子链中子节点数量增加，概率明显呈下降趋势。这种顺序评估展现了模型处理不断演变的信息线程的能力：模型能够根据累积的上下文动态调整概率，同时借助已建立的知识库保持评估一致性。完整的提示模板见附录H，以确保结果可完全复现。

为应对大语言模型增强图中虚拟边极少的极端情况（这可能源于模型知识库或推理能力的局限），我们提出一种缓解策略。该方法旨在减轻这些异常情况下大语言模型的不当影响，同时利用图学习方法纠正偏见（Li等，2025；Hang，2025）。具体而言，若虚拟边数量低于预设的最小阈值（表示连接稀疏），我们会保留与大语言模型所赋予的谣言概率排名靠前的若干子节点之间的连接，从而确保虚拟边达到基准数量，让图传播能够进一步优化并纠正大语言模型可能存在的误判。形式化地，对于子节点 $\{v_1, \ldots, v_n\}$ 及其对应的谣言概率 $P = \{p_1, \ldots, p_n\}$，原方法会在满足 $p_i > \theta$ 时，从虚拟节点向 v_i 添加一条虚拟边。然而，若最终边数 $|E_v| < \lceil \gamma n \rceil$，我们便按降序排列 P，并将虚拟节点连接到排名前 $k = \lceil \gamma n \rceil$ 的子节点，即使 $p_i \leq \theta$。这可以表示为：

$$E_v = \begin{cases} \{(v_i, v^*) \mid p_i > \theta\} & \text{if } |E_v| \geq \lceil \gamma n \rceil \\ \{(v_{\sigma(j)}, v^*) \mid 1 \leq j \leq k\} & \text{otherwise.} \end{cases}$$

此处，v^* 表示虚拟节点，σ 表示按 $p_{\sigma(1)} \geq \cdots \geq p_{\sigma(n)}$ 对索引排序的排列，$k = \lceil \gamma n \rceil$ 确保至少保留一定比例的边。在附录D中，我们提供了一项消融研究，表明 $\gamma = 0.2$ 能够持续获得稳定的性能；在相同的骨干网络下，根节点—虚拟节点链接预测的表现优于仅根节点分类。附录E进一步分析了虚拟节点连通性和深度效应，附录F则报告了子链词元统计，以刻画实际输入规模。这两部分内容进一步支持了我们方法的稳健性和良好性能。

为了证明　　为展示我们的框架对图神经网络的增强效果，我们将以双向图

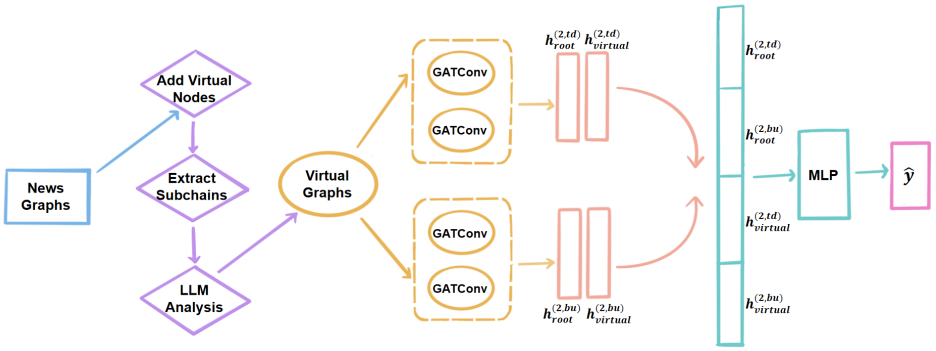


图 4：大语言模型增强的 Bi-GAT 模型。

注意力网络（Bi-GAT）为例。为使用虚拟节点实现 Bi-GAT，将给定的图 $\mathbb{G}=(\mathbb{V},\mathbb{E})$ 扩展为 $\mathbb{G}=(\mathbb{V},\mathbb{E})$，其中 $\mathbb{V}=\mathbb{V}\cup\{v^*\}$ 和 $\mathbb{E}=\{(v_i,v_j),(v_j,v_i):(v_i,v_j)\in\mathbb{E}\}\cup\mathbb{E}'$。

图注意力卷积 (GATConv) GATConv 函数实现了 Veličković 等人（2018年）提出的图注意力机制，该机制通过~ ~ ~关注邻近节点并利用学习得到的注意力系数来计算节点表示。对于具有节点特征 $\mathbf{x}_i\in\mathbb{R}^F$ 的图 $\mathbb{G}=(\mathbb{V},\mathbb{E})$，节点 $v_i\in\tilde{\mathbb{V}}$ （ v^*的节点特征设为零）的 GATConv 层~会计算其邻居节点 $v_j\in\mathcal{N}(i)$ 的注意力分数，其中 $\mathcal{N}(i)=\{v_j\,|\,(v_i,v_j)\in\mathbb{E}\}$ 是邻近节点的集合。节点 v_i 和v_j 之间的注意力分数 e_{ij} 计算如下：

$$e_{ij}=\text{LeakyReLU}\left(\mathbf{a}^T[\mathbf{W}_q\mathbf{x}_i\parallel\mathbf{W}_k\mathbf{x}_j]\right),$$

其中，$\mathbf{W}_q$、$\mathbf{W}_k$、$\mathbf{W}_v\in\mathbb{R}^{F,\times F}$是将输入特征转换到隐藏维度的权重矩阵。$\mathbf{a}\in\mathbb{R}^{2F'}$是注意力参数向量。 $[\mathbf{W}_q\mathbf{x}_i\parallel\mathbf{W}_k\mathbf{x}_j]\in\mathbb{R}^{2F'}$表示节点 v_i 和v_j的转换后特征的拼接。随后，使用 softmax 函数对邻居节点的注意力系数 α_{ij} 进行归一化，并利用注意力系数对邻居值向量加权，以更新节点 i的特征：

$$\alpha_{ij}=\frac{\exp(e_{ij})}{\sum_k\exp(e_{ik})},\quad \mathbf{h}_i=\sum_j\alpha_{ij}\mathbf{W}_v x_j,$$

其中 $k,j\in\mathcal{N}(i)$。为稳定并增强注意力机制，采用多个注意力头。每个注意力头 $h=1,\ldots,H$都有自己的权重矩阵 $\mathbf{W}_v^{(h)}\in\mathbb{R}^{F'\times F}$和注意力向量$\mathbf{a}^{(h)}\in\mathbb{R}^{2F'}$。每个注意力头的输出为：

$$\mathbf{h}_i^{(h)}=\sum_{j\in\mathcal{N}(i)}\alpha_{ij}^{(h)}\mathbf{W}_v^{(h)}\mathbf{x}_j,$$
$$\mathbf{h}_i=\|_{h=1}^{H}\mathbf{h}_i^{(h)}\in\mathbb{R}^{HF'},$$

其中 $\alpha_{ij}^{(h)}$ 是第 h 个头的注意力系数。$\mathbf{h}_i\in\mathbb{R}^{HF'}$ 是通过拼接各头输出得到的节点 v_i的最终输出。

自顶向下图注意力网络（TD-GAT）模拟图中从“高层”节点到“低层”节点的信息传递，适合捕捉因果关系。

- **第一层 GATConv**:

$$\mathbf{h}_i^{(1,\mathrm{td})} = \mathrm{ReLU}\left(\|_{h=1}^{H} \sum_{j \in \mathcal{N}_{out}(i)} \alpha_{ij}^{(h)} \mathbf{W}_{\mathrm{td1}}^{(h)} \mathbf{x}_j \right),$$

其中，$\mathcal{N}_{out}(i)$ 是节点 i 的出邻居，$\mathbf{h}_i^{(1,\mathrm{td})} \in \mathbb{R}^{HF'}$。

- **第二层 GATConv**:

$$\mathbf{h}_i^{(2,\mathrm{td})} = \frac{1}{H} \sum_{h=1}^{H} \sum_{j \in \mathcal{N}_{out}(i)} \alpha_{ij} \mathbf{W}_{\mathrm{td2}} [\mathbf{h}_j^{(1,\mathrm{td})} \parallel \mathbf{x}_{\mathrm{r}}],$$

其中，x_r 是根节点特征，$\mathbf{h}_i^{(2,\mathrm{td})} \in \mathbb{R}^{F'}$。

自底向上图注意力网络（BU-GAT）模拟特征从低层节点向高层节点的聚合。

- **第一层 GATConv**:

$$\mathbf{h}_i^{(1,\mathrm{bu})} = \mathrm{ReLU}\left(\|_{h=1}^{H} \sum_{j \in \mathcal{N}_{in}(i)} \alpha_{ij}^{(h)} \mathbf{W}_{\mathrm{bu1}}^{(h)} \mathbf{x}_j \right),$$

其中$\mathcal{N}_{in}(i)$ 是节点 i 的入邻居。

- **第二个 GATConv 层**:

$$\mathbf{h}_i^{(2,\mathrm{bu})} = \frac{1}{H} \sum_{h=1}^{H} \sum_{j \in \mathcal{N}_{in}(i)} \alpha_{ij} \mathbf{W}_{\mathrm{bu2}} [\mathbf{h}_j^{(1,\mathrm{bu})} \parallel \mathbf{x}_{\mathrm{r}}].$$

特征融合与边分类

- **特征提取**：提取$\mathbf{h}_{(2,\ \mathrm{td})}$ 根节点，$\mathbf{h}_{(2,\ \mathrm{td})}$ 虚拟节点，$\mathbf{h}_{(2,\ \mathrm{bu})}$ 根节点，以及 $\mathbf{h}_{(2,\ \mathrm{bu})}$ 虚拟节点。
- **融合**：拼接为$\mathbf{c} = [\mathbf{h}_{(2,\ \mathrm{td})}$ 根节点 ，$\mathbf{h}_{(2,\ \mathrm{bu})}$ 根节点 ，$\mathbf{h}_{(2,\ \mathrm{td})}$ 虚拟节点，$\mathbf{h}_{(2,\ \mathrm{bu})}$ 虚拟节点$] \in \mathbb{R}^{4F'}$。
- **分类**：$\mathbf{z} = \mathbf{W}_{\text{边}}\mathbf{c} + \mathbf{b}_{\text{边}}$，$\hat{y}_{\text{边}} = \sigma(\mathbf{z})$，其中$\mathbf{W}_{\text{边}} \in \mathbb{R}^{4F'}$，$\mathbf{b}_{\text{边}} \in \mathbb{R}$，且 σ 是 Sigmoid函数。

模型与训练我们采用二元交叉熵函数。对于批次大小为 N 的批次，损失函数定义如下:

$$\mathbf{Loss} = -\frac{1}{N} \sum_{i=1}^{N} \omega_i \big[y_i \log(\sigma(z_i)) + (1 - y_i) \log(1 - \sigma(z_i)) \big]$$

其中， z_i 是模型输出的正类 logit（即虚拟节点与根节点之间存在边时的 logit 分数）。 y_i 是源新闻的真实标签， ω_i 是根据 y_i的类别和正类_权重确定的权重。正类_权重设为负类与正类比例，以提高正类（少数类）样本的损失权重，优先确保其得到正确预测，并抵消类别不平衡。我们使用 Adam 优化器，通过反向传播训练参数，以优化 Bi-GAT 模型的所有组成部分。我们采用 7:1:2 的训练集/验证集/测试集划分，并仅使用验证数据选择 θ 。对于 PHEME，我们在完整数据集上进行整体分层抽样（而非按事件抽样），然后汇总来自全部五个事件的验证实例，通过 Youden 指数（J）选择一个共享阈值。对于没有事件划分的微博，我们同样采用全局划分，并选择一个基于验证集确定的阈值。超参数设置如下：学习率 =0.00005，权重衰减 $=$ 1e-3，丢弃率 $= 0.3$，训练轮次最多为 150。我们采用早停机制；训练期间持续监控验证集上的 F1 分数，若连续 20 轮未超过历史最佳值，则自动停止训练，从而防止模型过拟合。

5 实验

5.1 数据集和大语言模型

我们在 PHEME 数据集中的五个新闻事件上进行实验：《查理周刊》枪击案（《查理周刊》）；迈克尔·布朗遇害案（弗格森）；德国之翼9525号航班（德国之翼空难）；2014年渥太华枪击案（渥太华枪击案）；林特咖啡馆围困事件（悉尼人质事件）；以及微博数据集（Ma 等，2016），如表1所示。预处理后，我们移除了没有回复的图。为进行公平比较，所有基线方法均采用了相同的预处理。该实验使用 DeepSeek-V3 作为基础模型。使用 Qwen-Plus 得到的其他数值结果详见附录B和C。PHEME 和微博均为公开数据集。

新闻事件	非谣言	谣言	总计
《查理周刊》	1621	458	2079
弗格森	859	284	1143
德国之翼空难	231	238	469
渥太华枪击案	420	470	890
悉尼人质事件	699	522	1221
微博	2313	2351	4664

表1：PHEME 和微博数据集的统计信息。

5.2 评估指标

表2列出了评估分类模型性能的基本指标：准确率、精确率、召回率和 F1 分数。

我们将所提出的框架命名为 LLM-VN，并将其与现有方法进行比较，作为对比基线：Bi-GCN（Bian等人，2020年）捕捉树中的双向依赖关系；RvNN（Socher等人，2011年）递归地将子节点组合为父节点；图注意力网络（Veličković等人，2018年）利用注意力衡量邻居的重要性；HD-TRANs（Ma等人，2020年）将 Transformer 与图网络相结合，用于动态图；LINE（Tang等人，2015年）提供大规模静态网络嵌入；DDGCN（Korban和Li，2020年）对动态有向图中的时空特征进行建模；GraphSAGE（Hamilton等人，2017年）通过归纳方式采样并聚合局部特征；

指标	公式	Term
准确率	$\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}$	TP
精确率	$\frac{\text{TP}}{\text{TP}+\text{FP}}$	TN
召回率	$\frac{\text{TP}}{\text{TP}+\text{FN}}$	FP
F1 分数	$2 \cdot \frac{\text{精确率} \cdot \text{召回率}}{\text{精确率}+\text{召回率}}$	FN

表 2：分类指标、术语和公式。

GSMA（Ma 等人，2024a）通过引入注意力和位置编码增强 GraphSAGE，以进行谣言检测。SePro（Zeng 等人，2025）通过语义传播协作和线索链提示优化社交语境，从而提升大语言模型的性能。

上述基线方法的“LLM-VN”增强版本记为“**LLM-VN+基线方法**”。我们还将本方法与简单的模型平均进行比较：后者将大语言模型在精心设计的提示下得到的谣言概率，与基线方法得到的概率进行加权平均。最优权重通过线搜索进行调整。我们将其记为“**大语言模型+基线方法**”。

5.3 结果与分析

如表 3 所示，在 PHEME 数据集上，如果没有经过精心设计的提示，大语言模型在谣言识别方面表现不佳。准确率（谣言）衡量所有真实谣言实例中被正确识别的比例，而准确率（非谣言）则对“非谣言”实例进行同样的衡量。

新闻事件	准确率（谣言）	准确率（非谣言）
《查理周刊》	15.54%	83.20%
弗格森	41.55%	89.76%
德国之翼空难	23.11%	91.77%
渥太华枪击案	16.81%	87.62%
悉尼人质事件	33.14%	78.25%

表3：大语言模型在基础提示下的性能。

我们提出的框架“LLM-VN”与 Bi-GAT 结合后在 PHEME 数据集上的表现如表4所示：

新闻事件	**Acc.**	**Pre.**	**Rec.**	**F1**	**AUC**
《查理周刊》	0.923	0.943	0.967	0.955	0.962
弗格森	0.860	0.818	0.892	0.853	0.903
德国之翼空难	0.894	0.914	0.961	0.937	0.827
渥太华枪击案	0.876	0.883	0.987	0.932	0.846
悉尼人质事件	0.829	0.866	0.905	0.885	0.858

表4：LLM-VN+Bi-GAT 的性能。

从表4可以看出，该模型在大多数新闻事件上均表现出色，准确率通常超过85%，且精确率、召回率和F1分数也较高。我们在PHEME上的结果表明，即使回复证据相对稀少，该框架仍然有效。为更好地说明我们方法与模型无关的特性，并展示虚拟节点相较于提示工程带来的额外优势，我们在表5中全面比较了在微博和PHEME数据集上用于谣言检测的图学习方法。评估涵盖三种设置下的基线模型：原始基线、LLM+Baseline和LLM-VN+Baseline。为确保比较公平，

	PHEME 数据集					微博数据集				
方法	**Acc**	**Prec**	**Rec**	**F1**	**类别准**	**确率**	**Prec**	**Rec**	**F1**	**类别**
Bi-GCN	0.824	0.753/0.861	0.734/0.872	0.741/0.865	R/N	0.963	0.948/0.970	0.946/0.972	0.947/0.971	R/N
大语言模型+Bi-GCN	0.830	0.758/0.862	0.744/0.850	0.750/0.855	R/N	0.971	0.962/0.975	0.959/0.979	0.960/0.976	R/N
LLM-VN+Bi-GCN	**0.842**	0.772/0.872	0.752/0.890	0.761/0.884	R/N	**0.988**	0.976/0.992	0.978/0.991	0.976/0.992	R/N
RvNN	0.763	0.689/0.796	0.587/0.858	0.631/0.825	R/N	0.771	0.723/0.782	0.681/0.795	0.646/0.817	R/N
大语言模型+RvNN	0.783	0.777/0.811	0.754/0.802	0.761/0.808	R/N	0.775	0.692/0.793	0.595/0.800	0.638/0.796	R/N
LLM-VN+RvNN	**0.807**	0.773/0.835	0.696/0.896	0.784/0.842	R/N	**0.781**	0.773/0.791	0.796/0.766	0.784/0.778	R/N
GraphSAGE	0.842	0.772/0.876	0.820/0.878	0.795/0.877	R/N	0.963	0.956/0.972	0.953/0.975	0.954/0.973	R/N
GSMA	0.848	0.834/0.856	0.823/0.851	0.840/0.860	R/N	0.974	0.967/0.973	0.953/0.982	0.960/0.977	R/N
LLM-VN+GraphSAGE	**0.866**	0.822/0.887	0.825/0.879	0.820/0.892	R/N	**0.981**	0.978/0.972	0.933/0.992	0.955/0.982	R/N
GAT	0.811	0.733/0.877	0.541/0.798	0.405/0.833	R/N	0.947	0.939/0.961	0.936/0.943	0.938/0.962	R/N
大语言模型+图注意力网络	0.823	0.799/0.826	0.796/0.820	0.804/0.828	R/N	0.960	0.944/0.969	0.939/0.953	0.941/0.961	R/N
SePro	0.831	0.778/0.852	0.762/0.864	0.770/0.858	R/N	0.950	0.947/0.954	0.948/0.953	0.947/0.953	R/N
LLM-VN+图注意力网络	**0.847**	0.823/0.868	0.860/0.850	0.800/0.870	R/N	**0.982**	0.977/0.987	0.972/0.985	0.981/0.989	R/N
HD-TRANs	0.766	0.656/0.783	0.697/0.755	0.676/0.768	R/N	0.974	0.957/0.979	0.946/0.978	0.952/0.979	R/N
大语言模型+HD-TRANs	0.779	0.767/0.784	0.766/0.782	0.780/0.777	R/N	0.980	0.961/0.984	0.952/0.988	0.959/0.987	R/N
LLM-VN+HD-TRANs	**0.796**	0.696/0.811	0.737/0.792	0.716/0.802	R/N	**0.991**	0.963/0.981	0.960/0.984	0.962/0.989	R/N
LINE	0.744	0.732/0.749	0.730/0.750	0.733/0.749	R/N	0.790	0.763/0.802	0.771/0.811	0.760/0.796	R/N
大语言模型+LINE	0.759	0.753/0.760	0.750/0.758	0.755/0.763	R/N	0.803	0.768/0.804	0.775/0.806	0.763/0.803	R/N
LLM-VN+LINE	**0.786**	0.747/0.794	0.745/0.782	0.746/0.797	R/N	**0.811**	0.802/0.820	0.804/0.816	0.800/0.823	R/N
DDGCN	0.855	0.877/0.831	0.763/0.892	0.816/0.860	R/N	0.948	0.941/0.965	0.933/0.970	0.937/0.967	R/N
大语言模型+DDGCN	0.860	0.855/0.866	0.852/0.861	0.856/0.867	R/N	0.954	0.950/0.959	0.949/0.955	0.952/0.963	R/N
LLM-VN+DDGCN	**0.876**	0.858/0.861	0.832/0.867	0.845/0.864	R/N	**0.984**	0.979/0.988	0.978/0.985	0.982/0.990	R/N

表5：PHEME 和微博数据集上的性能比较。

我们提出的提示策略（附录H）应用于所有评估的基线方法。我们遵循PHEME和微博上的标准基准设置，而非严格基于时间的早期检测协议。在我们的框架中，预训练大语言模型知识仅作为辅助信号，用于增强基于图神经网络的谣言检测，而最终预测仍由下游图模型作出。从表5可以看出，LLM-VN增强方法在PHEME和微博数据集上均展现出显著优势。与原始基线方法和模型平均方法（LLM+Baseline）相比，我们的框架在准确率、精确率、召回率和F1分数等关键指标上均有明显提升。尤其是在PHEME数据集上，LLM-VN+DDGCN等方法在准确率和F1分数方面表现尤为突出。在微博数据集上，LLM-VN+Bi-GCN和LLM-VN+GAT等方法也优于基线，尤其在非谣言类别的精确率和召回率方面表现更佳，体现出更高的分类稳定性。相比之下，LLM+Baseline的提升不如其VN增强版本稳健，尤其是在传播结构复杂或数据分布不平衡的情况下。在微博数据集上，各方法的表现优于PHEME，这得益于微博拥有规模更大、层次更深且内容更丰富的传播树。总体而言，LLM-VN框架通过融合大语言模型的语义理解能力与图模型的结构化处理能力，提升了泛化能力和分类准确率，尤其适用于复杂的社交媒体传播模式。

6 结论

我们提出一个通用框架，通过利用大语言模型来提升图学习方法在谣言预测上的性能。通过使用大语言模型分析子链并分配谣言概率，我们用一个虚拟“是谣言”节点来增强该图。直观而言，当大语言模型预测

在许多子链上给出较高的谣言概率时，该虚拟节点会与根节点共享更多邻居节点，从而在图神经网络传播过程中产生趋于一致的嵌入。这提高了根节点与虚拟节点之间链接预测的可能性，从而有效地将源新闻分类为谣言。我们设计的一个关键优势在于其对不完美大语言模型信号的鲁棒性：缓解机制降低了对噪声或弱证据的过度依赖，例如早期扩散阶段或稀疏且带有偏见的回复。值得注意的是，我们的方法是一个模块化框架，它利用大语言模型检索并推理子链证据，并将该信号注入传播图，以进行基于图神经网络的预测。这种与模型无关的设计使其能够与不同的图神经网络骨干网络和大语言模型搭配使用。随着图神经网络和大语言模型的持续进步，我们的框架也将随之自然提升。此外，如附录 A 所示，我们方法中大语言模型的集成仅依赖 API 调用，从而确保较低的计算成本。对于 PHEME 数据集（节点数少于 100），每条新闻的处理时间通常约为 10 秒；对于微博数据集（节点数达数千），则为 30–60 秒。

局限性

处理超大规模图时，我们的方法会显现出局限性。尽管附录 A 中分析了成本和运行时间，但依赖大语言模型 API 仍可能造成不可忽视的延迟，导致无法对超大规模图实现完全实时的响应。

参考文献

Tian Bian、Xiao Xi、Tingyang Xu、Peilin Zhao、Wenbing Huang、Yu Rong 和 Junzhou Huang。2020。基于双向图卷积网络的社交媒体谣言检测。载于人工智能促进协会会议论文集（第 34 卷，第 1 期，第 549–556 页）。AAAI出版社。doi:10.1609/aaai.v34i01.5393。

Tsun Hin Cheung 和 Kin ManLam。2023。FactLLaMA：利用外部知识优化指令遵循语言模型以实现自动事实核查。载于*2023*年亚太信号与信息处理协会年度峰会暨会议（*APSIPA ASC*）（第846–853 页）。IEEE。doi:10.1109/APSIPAASC58517.2023.10317251。

William L. Hamilton、Rex Ying 和 Jure Leskovec。2017年。《大图上的归纳式表示学习》。载于《神经信息处理系统进展》第*30*卷（第30卷，第1024-1034页）。Curran Associates。

Ching Nam Hang、Peiduo Yua 和 Chee Wei Tan。2025年。TrumorGPT：基于图检索增强的大语言模型事实核查。*arXiv:2505.07891*。

Tianyi Huang、Jingyuan Yi、Peiyang Yu 和 Xiaochuan Xu。2025年。揭开数字虚假信息的面纱：基于大语言模型的错误信息检测策略比较分析。*arXiv:2503.00724*。

Matthew Korban 和 Xin Li。2020。DDGCN：用于动作识别的动态图卷积网络。载于《欧洲计算机视觉大会》（第761-776页）。施普林格。doi:10.1007/978-3-030-58565-5_45。

Sumeet Kumar 和 Kathleen Carley。2019。结合卷积单元的树形长短期记忆网络，用于预测社交媒体对话中的立场和谣言真实性。载于第*57*届年会论文集

计算语言学协会（第 . 5047-5058）. 计算语言学协会
语言学。doi:10.18653/v1/P19-1498。Jianqiao Lai、Xinran Yang、Wenyue Luo、Linjiang Zhou、Langchen Li、Yongqi Wang 和 Xiaochuan Shi。2024年。RumorLLM：一种基于谣言大语言模型的假新闻检测数据增强方法。《应用科学》，14(8):3532。doi:10.3390/app14083532。Haitao Li、Qian Dong、Junjie Chen、Huixue Su、Yujia Zhou、Qingyao Ai、Ziyi Ye 和 Yiqun Liu。2024年。以大语言模型为评判者：关于基于大语言模型的评估框架的全面综述。*arXiv:2412.05579*Guoyi Li、Die Hu、Zongzhen Liu、Xiaodan Zhang、Honglei Lyu。2025年。利用大语言模型和异构图自编码器进行语义重排，以增强谣言检测。载于第*31*届国际计算语言学会议论文集 （第8557-8572页）。计算语言学协会。doi:10.18653/v1/2025.coling-main.572。

Tianrui Liu、Qi Cai、Changxin Xu、Bo Hong、Fanghao Ni、Yuxin Qiao 和 Tsungwei Yang。2024年。《一种新型图神经网络方法用于谣言检测》。《style id='1'>《科学技术学术期刊》，10(1)：1-6。doi:10.54097/farmdr42。

Jing Ma、Wei Gao、Prasenjit Mitra、Sejeong Kwon、Bernard J.Jansen、Kam-Fai Wong 和 Meeyoung Cha。2016 年。《利用循环神经网络检测微博中的谣言》。载于 第二十五届国际人工智能联合会议论文集（*IJCAI 2016*）（第3818-3824页）。AAAI出版社。

Jing Ma、Wei Gao 和 Kam-Fai Wong。2017年。《利用核学习通过传播结构检测微博帖子中的谣言》。载于 计算语言学协会第*55*届年会论文集（第*1*卷：长篇论文）（第708-717页）。计算语言学协会。doi:10.18653/v1/P17-1066。

Jing Ma 和 Wei Gao。2020年。使用树形 Transformer 反驳 Twitter 上的谣言。载于 第 *28* 届国际计算语言学会议（*COLING*）论文集 （第 5455-5466 页）。计算语言学协会。doi:10.18653/v1/2020.coling-main.476。

Manfu Ma、Cong Zhang、Yong Li、Jiahao Chen 和 Xuegang Wang。2024年。聚焦节点位置的加权 GraphSAGE 谣言检测模型。《科学报告》，14(1):27127。doi:10.1038/s41598-024-76738-7。Xiaoxiao Ma、Yuchen Zhang、Kaize Ding、Jian Yang、Jia Wu 和 Hao Fan。2024年。利用大语言模型增强的语义挖掘进行假新闻检测。载于 *2024*年自然语言处理实证方法会议论文集 （第 508-521 页）。计算语言学协会。doi:10.18653/v1/2024.emnlp-main.31。Suvendu Mohanty。2025年。大语言模型中的细粒度偏见检测：增强对细微偏见的检测机制。*arXiv:2503.06054*。Dahlia Shehata、Robin Cohen 和 Charles Clarke。2024年。使用超大规模语言模型进行谣言评估。*arXiv:2404.16859*。

Richard Socher、Cliff Chiung-Yu Lin、Andrew Y. Ng 和 Christopher D. Manning。2011。使用递归神经网络解析自然场景和自然语言。载于第*28*届国际机器学习大会论文集 （第129-136页）。Omnipress。

Tiening Sun、ZhongQian、董素君、PeifengLi 和 QiaomingZhu。2022年。基于图对抗对比学习的社交媒体谣言检测。载于*ACM Web* 会议 *2022*年论文集 （第2789-2797页）。ACM。doi:10.1145/3485447.3511999。

唐健、屈萌、王明哲、张明、严军和梅巧珠。2015年。LINE：大规模信息网络嵌入。载于第*24*届国际万维网大会（*WWW*）论文集（第1067–1077页）。ACM。doi:10.1145/2736277.2741093。

田志亮、黄景源、何泽江、黄震、陆梦龙、乔林波、梅松竹、王一杰和李东升。2025年。基于影响力引导的样本选择和基于博弈的视角分析的大语言模型谣言检测。载于第*61*届计算语言学协会年会论文集（第1378–1391页）。计算语言学协会。doi:10.18653/v1/2025.acl-long.1378。

Petar Veličković、Guillem Cucurull、Arantxa Casanova、Adriana Romero、Pietro Liò和Yoshua Bengio。2018年。图注意力网络。载于国际学习表征会议。OpenReview.net。

王博、Jing Ma、林宏展、杨志伟、杨瑞超、田源和常毅。2024年。通过竞争智慧防御，利用大语言模型实现可解释的假新闻检测。载于*ACM Web* 会议 *2024*年论文集（第2452–2463页）。ACM。doi:10.1145/3589334.3645471。

ZhiyuanWu、皮德昌、JunfuChen、谢萌和JianjunCao。2020。基于带注意力机制的传播图神经网络的谣言检测。《*Expert Systems with Applications*》，158：113595。doi:10.1016/j.eswa.2020.113595。doi:10.1016/j.eswa.2020.113595。

严叶青、郑鹏和王永军。2024年。通过知识驱动提示增强大语言模型的谣言检测能力。《人工智能工程应用》，133：108259。doi:10.1016/j.engappai.2024.108259。

曾怡蓉、丁晓、蔡碧波、刘挺和秦兵。2025年。内容审核至关重要：探索大语言模型在社交媒体上的有效谣言检测。载于*2025*年计算语言学协会北美分会会议论文集：人类语言技术（第*1*卷：长论文）（第2537–2552页）。计算语言学协会。

Chunting Zhou、Pengfei Liu、Puxin Xu、Srini Iyer、Jiao Sun、Yuning Mao、Xuezhe Ma、Avia Efrat、Ping Yu、Lili Yu、Susan Zhang、Gargi Ghosh、Mike Lewis、Luke Zettlemoyer和Omer Levy。2023年。LIMA：少即是多的对齐。载于 第*36*届神经信息处理系统进展。Curran Associates。

Mohammad Ziari和Nasrollah Moghaddam Charkari。2025年。社交网络上的谣言检测与传播：综述。《*Expert Systems with Applications*》，263:128798。doi:10.1016/j.eswa.2025.128798。

Arkaitz Zubiaga、Ahmet Aker、Kalina Bontcheva、Maria Liakata 和 Rob Procter。2018年。《社交媒体谣言的检测与消解：一项综述》。《ACM计算调查》，51(2):1-36。doi:10.1145/3161603。

计算成本

在单线程下，通义千问-Plus 和 DeepSeek-V3 每小时处理的节点数均超过 1,200 个，通过多线程还可以进一步加速。由于响应仅限于概率，令牌用量和成本都保持在较低水平。

在标准工作站（$\geq$16 核心、 $\geq$32 GB 内存、稳定的 $\geq$100 Mbps 网络）上，使用简单的异步多线程（向 DeepSeek-V3 或 Qwen-Plus 发起 100 个并行 API 调用），我们测量了处理这两个数据集的实际计算时间和 API 成本。如表 6 所示，这些结果证明了我们的框架在资源受限环境中的实际部署能力。PHEME 数据集中的图通常只有几十个节点，处理每条新闻条目约需 10 秒；而微博数据集中的图通常包含数千个节点，处理每条新闻条目需 30–60 秒。

指标	PHEME	微博
树	5,447	3,805
帖子	96,344	3,804,357
时间（100线程）	40–50分钟	35–40小时
API成本（美元）	5–10	80–100

表 6：标准工作站上大语言模型的计算成本。

B 大语言模型偏见

为进一步展示大语言模型固有的偏见，我们使用通义千问-Plus评估了它们在 PHEME 数据集上的表现。表 7 的分析显示，与 DeepSeek 相比，通义千问表现出显著不同的偏见。这种差异主要源于两者不同的训练数据和强化学习策略。具体而言，通义千问的立场更为主观且激进，倾向于谨慎行事，优先避免假阴性，即使这会导致假阳性增加。相比之下，DeepSeek 的做法则更加保守、中立，倾向于尽量减少假阳性，而非冒假阴性的风险。

新闻事件	准确率（谣言）	准确率（非谣言）
《查理周刊》	92.3%	21.8%
弗格森	88.7%	34.5%
德国之翼空难	94.1%	19.2%
渥太华枪击案	90.6%	26.7%
悉尼人质事件	86.9%	31.4%

表 7：Qwen-Plus 在 PHEME 上使用基础提示时的性能。

C 模型无关性主张

我们使用 Qwen-Plus 在 PHEME 数据集上开展了额外实验，以强化我们的模型无关性主张。如表 8所示，我们的框架在各基线方法上均带来一致提升，准确率增益为 1-3%，R/N 类别的 F1 增益为 2-5%，与使用 DeepSeek-V3 所得结果高度吻合。我们对两种大语言模型均采用默认温度，且未

进行验证调参：DeepSeek 为 1.0，Qwen-Plus 为 0.7。实践中，默认设置已足够，并可避免引入另一个解码超参数。这些结果证实了该框架在不同大语言模型系列上的稳健性。

方法	Acc	Prec	Rec	F1	类别
Bi-GCN	0.824	0.753/0.861	0.734/0.872	0.741/0.865	R/N
LLM-VN+Bi-GCN	**0.833**	0.752/0.880	0.778/0.858	0.767/0.864	R/N
RvNN	0.763	0.689/0.796	0.587/0.858	0.631/0.825	R/N
LLM-VN+RvNN	**0.786**	0.773/0.835	0.696/0.896	0.784/0.842	R/N
Graphsage	0.842	0.772/0.876	0.820/0.878	0.795/0.877	R/N
GSMA	0.848	0.834/0.856	0.823/0.851	0.840/0.860	R/N
LLM-VN+GraphSAGE	**0.852**	0.803/0.882	0.821/0.871	0.814/0.876	R/N
GAT	0.811	0.733/0.877	0.541/0.798	0.405/0.833	R/N
LLM-VN+图注意力网络	**0.823**	0.730/0.880	0.780/0.850	0.750/0.860	R/N
HD-TRANs	0.766	0.656/0.783	0.697/0.755	0.676/0.768	R/N
LLM-VN+HD-TRANs	**0.779**	0.712/0.808	0.729/0.790	0.722/0.800	R/N
LINE	0.744	0.732/0.749	0.730/0.750	0.733/0.749	R/N
LLM-VN+LINE	**0.755**	0.702/0.782	0.727/0.769	0.720/0.776	R/N
DDGCN	0.855	0.877/0.831	0.763/0.892	0.816/0.860	R/N
LLM-VN+DDGCN	**0.865**	0.811/0.890	0.832/0.877	0.82/0.881	R/N

表8：基于通义千问-Plus 的 PHEME 数据集性能比较。

D 消融研究

我们在 PHEME 数据集上，使用通义千问-Plus ＋ Bi-GAT 对 γ 进行了消融研究。表9中的结果表明，$\gamma = 0.2$ 始终取得最佳性能，在融入全局上下文与保留局部结构信号之间实现了最佳平衡。

γ (%)	Acc.	Rec.	F1	精确率	AUC
10	0.8138	0.7727	0.7401	0.7101	0.8767
15	0.8083	0.8021	0.7417	0.6897	0.8780
20	**0.8229**	**0.7754**	**0.7503**	**0.7268**	**0.8798**
25	0.8128	0.7888	0.7431	0.7024	0.8763
30	0.8018	0.8155	0.7385	0.6748	0.8744

表9：虚拟节点混合系数 γ的消融研究。

我们进一步在相同的骨干网络和训练流程下，对两种预测形式进行对照实验。具体而言，我们比较了（i）仅根节点分类，即直接根据根节点表示预测谣言标签；以及（ii）根节点—虚拟节点链接预测，即在虚拟边增强后预测根节点是否与虚拟节点相连。由于编码器和数据处理保持一致，这一比较可以直接检验链接预测形式是否比仅根据根节点进行分类更有效。

E 虚拟节点分析

我们从两个角度对所提出的虚拟节点增强进行了补充分析。首先，我们在表11中报告了 PHEME 各事件中虚拟节点的度分布（不同邻居节点的数量），以量化虚拟节点连通性。p75 值表明，在典型情况下，虚拟节点只连接到数量适中的节点，而非形成全连接枢纽；最大值则凸显了少数规模较大的图，其连通性

事件	仅根节点准确率	链接预测准确率
《查理周刊》	0.905	0.923
弗格森	0.838	0.860
德国之翼空难	0.867	0.894
渥太华枪击案	0.871	0.876
悉尼人质事件	0.801	0.829

表10：仅根节点分类与根节点—虚拟节点链接预测（准确率）

事件	划分	p75	max	mean
德国之翼空难	ALL	4	47	3.96
	谣言	11	32	7.81
	非谣言	3	47	3.23
弗格森	ALL	11	217	9.14
	谣言	18	94	13.96
	非谣言	9	217	8.09
渥太华枪击案	ALL	2	85	2.06
	谣言	3	85	3.71
	非谣言	2	34	1.78
悉尼围城事件	ALL	12	341	9.36
	谣言	18	341	15.32
	非谣言	9	126	7.20
《查理周刊》	ALL	3	146	2.90
	谣言	8	93	6.40
	非谣言	2	146	2.21

表11：PHEME 各事件中的虚拟节点度统计。

可能更高。其次，我们考察大语言模型赋予的子链概率是否表现出系统性的深度/长度偏好。我们在表12中计算了节点深度与大语言模型赋予的概率之间的斯皮尔曼等级相关系数。接近于零的相关性表明，大语言模型评分并未持续偏向浅层或深层节点，这支持了我们的虚拟边选择并非受大语言模型评分中的深度偏见所驱动。

F 子链长度和词元统计

在大型数据集微博上，我们计算了所有图中所有节点的平均子链长度。为评估最坏情况下的输入规模，我们还找出了 4,664 个微博事件图中全局最长的子链，并统计沿该路径拼接所有节点文本后的词元数量。如表13所示，子链通常较浅。我们的输入长度远低于主流大语言模型 API 的上下文长度限制，例如，DeepSeek 支持 128K 个词元，而 GPT、通义千问和 Gemini 支持约 1M 个词元，Grok 则支持 2M 个词元。在我们的设置中，截断并非实际问题。

划分	Spearman ρ （深度，概率）
所有节点	0.0285
谣言（ref=0）	0.0060
非谣言（ref=1）	−0.0737

表12：深度偏差分析：节点深度与大语言模型赋予的概率之间的相关性。接近零的 ρ 表明大语言模型评分不存在系统性的深度/长度偏好。

统计数据（微博）	数值
平均子链长度（节点数）	2.687
最大子链长度（节点数）	73
最大子链上的词元（拼接）	8385

表13：微博子链长度和词元长度统计（全局最长子链：3907653488846171.json）。

G 大语言模型性能案例研究

为了说明使用基础提示在 PHEME 数据集上进行谣言检测时大语言模型的局限性，我们以《查理周刊》枪击事件中的一个典型案例为例。

- **向大语言模型提问：**“《查理周刊》枪击案是什么？”
- **大语言模型的回答：**

  “2015年1月7日上午约11:30，在法国巴黎，法国讽刺性周刊《查理周刊》的员工遭到两名出生于法国的阿尔及利亚裔穆斯林兄弟赛义德·库阿希和谢里夫·库阿希的恐怖主义枪击袭击。这两人携带步枪和其他武器，杀害了12人，另有11人受伤；他们自称是阿拉伯半岛基地组织成员，该组织声称对此次袭击负责。……”

- **来源推文（来自 PHEME 数据集）：**

  “巴黎《查理周刊》枪击事件造成10人死亡。”

- **向大语言模型发送的后续提示：**“这条推文是在传播谣言吗？”
- **大语言模型回复：**

  “不是，这指的是2015年1月7日在巴黎发生的真实《查理周刊》恐怖袭击。”

此案例表明，大语言模型对重大历史事件拥有丰富的背景知识，能够准确回忆日期、地点、作案者以及官方公布的死亡人数（12名受害者）等关键信息。然而，在评估这条具体推文时，大语言模型忽略了一个关键差异：推文称“10人死亡”，这反映了突发新闻阶段流传的一则不准确早期报道（在确认死亡人数为12人之前，早期媒体报道通常称有10或11人死亡）。

H 基于大语言模型的谣言评分提示模板

本附录展示了两阶段大语言模型评分流程中使用的完整提示模板。我们首先根据根帖构建一个稳定的背景知识库，然后使用每条根节点到节点子链，在相同判断依据下逐步更新谣言概率。

提示 A：根帖背景知识构建

```
You are evaluating the root post of a rumor-detection thread.

Your task is to build a stable background knowledge base and a reusable judgment
basis for later subchain evaluation.
Do NOT overcommit to a final rumor decision at this stage.

First identify the root post's key factual claims, then compare them against
general background knowledge. If you are uncertain,
 explicitly preserve uncertainty instead of inventing details.

Important principles:
(1) Separate factual assessment from stylistic suspicion.
(2) Treat direct contradiction with known facts as stronger evidence than emotional tone.
(3) Treat unsupported new key claims as moderate risk rather than conclusive proof of rumor.
(4) Emotional or forwarding-inducing language alone is only weak evidence.
(5) If the post is broadly plausible but lacks verification, do not assign an
overly confident non-rumor judgment.
(6) The output must define stable and reusable criteria that can be consistently
applied to later subchains.
(7) Do not assume that general familiarity with the event is sufficient evidence
that the post is true; evaluate the specific claims made in the post.

Please perform the following steps:

Step 1. Extract the key claims in the root post. For each claim,
 identify:- claim id- claim text
- claim type: event / casualty / time / place / entity / cause / other
- verifiability: high / medium / low

Step 2. Summarize the relevant background facts that are useful for evaluating
these claims. For each fact, identify:- fact text- relevance: high / medium / low
- certainty: high / medium / low

Step 3. Compare each extracted claim against the background fact. For each claim, determine:
- relation_to_background: consistent / contradicted / unsupported / uncertain
- severity: integer from 0 to 3- note: short explanation

Severity guidance:
```

```
- 3 = direct contradiction in key factual content (e.g., entity, time, place,
number, or event outcome)
- 2 = important unsupported claim or suspiciously specific new factual addition
without support- 1 = weak stylistic or rhetorical signal only
- 0 = no apparent issue

Step 4. Identify style-based risk signals without letting them dominate factual
    evidence.
Report:
- emotional_language: 0 or 1
- urgency_or_forwarding: 0 or 1
- missing_source_for_key_claim: 0 or 1

Step 5. Produce an initial rumor prior score for the root post.
Use a float between 0 and 1:
- 0 means strongly non-rumor
- 1 means strongly rumor
- 0.5 means uncertain / insufficient evidence

Scoring rule:
- Start from uncertainty when evidence is limited.
- Move upward mainly for factual contradiction or important unsupported claims.
- Move downward only when the post is broadly consistent with background facts and
     does not contain serious factual conflict.
- Avoid extreme scores unless the evidence is strong.

Return STRICT JSON ONLY. Do not use markdown fences. Do not output any extra text.

Required JSON schema:{"root_claims": [{"claim_id": "c1", "claim": "...",
"claim_type": "event|casualty|time|place|entity|cause|other"
"verifiability": "high|medium|low"}],"background_facts": [{"fact": "...",
"relevance": "high|medium|low", "certainty": "high|medium|low"}],"claim_checks": [
{"claim_id": "c1",
"relation_to_background": "consistent|contradicted|unsupported|uncertain",
"severity": 0, "note": "..."}],"style_risk_signals": {"emotional_language": 0,
"urgency_or_forwarding": 0, "missing_source_for_key_claim": 0},
```

```
   "root_prior_rumor_score": 0.5,
"reasoning_basis": "A concise reusable judgment basis for future subchain
scoring."}


Root post:
[ROOT_POST]
```

提示 B：基于子链的谣言概率增量更新

```
You are evaluating a root-to-node subchain in a rumor-detection graph.

You must strictly follow the provided background knowledge base and reasoning
    basis.
Do NOT create a new evaluation standard.
Your task is to update the rumor probability of the ROOT POST using the replies in
     this subchain.

The subchain contains the root post followed by replies in temporal order.
Assess how the later replies affect the credibility of the root post.

Important principles:
(1) Start from the root_prior_rumor_score in the provided knowledge base.
(2) Distinguish factual correction from mere tone or agreement. (3) Concrete evidence, contradiction,
  or correction matters more than emotion.
(4) Replies with no factual content should have little effect. (5) If the evidence is mixed or weak,
  keep the updated score near the prior rather than becoming overconfident.
(6) Do not assume that a widely known event automatically validates the specific
claims in the root post. (7) Evaluate the replies as incremental evidence for or against the root post.



Before producing the final score, internally determine whether each non-root reply
mainly functions as:
- support
- question
- deny
- correct
- add_evidence
- noise

Use these internal role judgments only to guide the probability update.
Do NOT output the role labels unless they are necessary for a brief explanation.

Update guidance:
- Increase rumor probability when a reply exposes factual inconsistency,
fabricated detail, unsupported new claim, or source-related credibility problems.
- Decrease rumor probability when a reply provides concrete correction, clarification,
  authoritative attribution, or evidence that resolves an apparent mismatch.
- Pure agreement or pure denial without evidence should have only weak influence.
- Emotional reactions without factual content should have minimal influence.
- If the subchain adds no meaningful new evidence, the final score should remain
close to the prior.



Final scoring rule:
```

```
- Begin from "root_prior_rumor_score".
- Apply only modest changes for weak or indirect evidence.
- Use larger changes only for concrete and relevant evidence.
- Keep the final score in [0, 1].
- Avoid extreme values unless the accumulated evidence is strong and consistent.

Return STRICT JSON ONLY. Do not use markdown fences. Do not output any extra text.

Required JSON schema:{"probability": 0.5,
"explanation": "A brief explanation of how the subchain updates the root rumor
score."}



Background knowledge base:
[KNOWLEDGE_BASE]

Subchain text:
[SUBCHAIN]
```

设计依据

- 我们首先根据根帖构建稳定的先验判断依据，然后将其应用于所有子链，从而确保不同子链在一致的标准下进行评分。
- 我们明确区分事实矛盾、无依据的主张和弱风格信号，避免表面修辞主导判断。
- 我们将子链评分建模为基于先验分数的增量更新，而非针对每个子链独立重新决策。
- 我们要求模型在内部区分每条回复的作用（例如支持、纠正、证据补充或噪声），但只输出最终概率和简要解释，从而在保留角色感知推理的同时降低词元成本。
- 当证据较弱时，我们保留不确定性，从而在虚拟边构建的阈值化之前提高分数稳定性。