DuoKD: Dual Knowledge Distillation from Large Language Models for Robust Graph Neural Networks

Cuiying Huo[1], Xiaotong Huang[2], Dongxiao He[1], Yixuan Du[1], Wenhuan Lu[1,3,*] Di Jin[1,3,*]

[1]College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]School of Mathematics, Tianjin University, Tianjin, China
[3]School of Intelligence Science and Engineering, Qinghai Minzu University, Xining, China
{huocuiying, huangxt_3604, hedongxiao, wenhuan, jindi}@tju.edu.cn

Abstract

Graph neural networks (GNNs) have become a dominant modeling paradigm for graph-structured data, and the emergence of large language models (LLMs) has spurred growing interest in integrating external semantic knowledge into GNNs. Current LLM-based GNNs are devoted to extracting semantically similar information from LLMs to enhance representation learning. However, they generally overlook key signals that are semantically dissimilar but exhibit stronger inter-class discriminative ability. Especially when the original graph data contains noise or semantic ambiguity, a single similarity-based semantic augmentation strategy not only fails to provide effective enhancement, but may also amplify misleading signals generated by the LLM in response to low-quality inputs or its own hallucinations, further degrading the discriminative power and robustness of GNNs. To this end, we propose a dual positive-negative knowledge extraction strategy based on LLMs, and integrate it with a knowledge distillation mechanism to dynamically transfer multi-dimensional enhanced signals to GNNs, thereby achieving fine-grained and robust graph representation learning. Specifically, we design personalized prompts to guide LLMs in generating semantically similar positive signals and semantically dissimilar negative signals, which help the model capture intra-class consistency and inter-class distinction. Then, we further generate structural and semantic reasoning as supplementary knowledge to support the rationality and guidance of supervision signals. To identify high-confidence transferred knowledge, we introduce a language-based evaluation mechanism to filter low-confidence or hallucinated outputs. Finally, under a unified distillation framework, our method uses both positive and negative knowledge to guide GNN training, achieving adaptive and robust representation learning. Extensive experiments on benchmark datasets verify the superior performance of our approach across various tasks.

Introduction

Graph-structured data are pervasive across real-world applications, including social networks (Fan et al. 2019; Jia et al. 2025; Wang et al. 2018, 2017), protein–protein interaction networks (Manipur et al. 2021), and transportation systems (Dai et al. 2020; Jin et al. 2025). Such data encode intricate relational dependencies among entities and have emerged as a fundamental focus in machine learning and data mining (Wang et al. 2020). Graph Neural Networks (GNNs) (Zhou et al. 2020; Wu et al. 2020; Huo et al. 2023), as a core framework for modeling graph data, effectively integrate both structural topology and node-level attributes within a unified deep learning paradigm. By leveraging message passing and neighborhood aggregation, GNNs iteratively refine node representations, enabling them to capture both local and global patterns. Due to their modeling capacity and versatility, GNNs have achieved remarkable success in a wide range of tasks, including node classification (Xiao et al. 2022), link prediction (Zhang and Chen 2018; Cai et al. 2021), and recommendation systems (Wu et al. 2022; Gao et al. 2023). Moreover, their applicability has been increasingly extended to broader domains such as natural language processing (Wu et al. 2023), computer vision (Chen et al. 2024a), and bioinformatics (Zhang et al. 2021), establishing GNNs as indispensable components in intelligent systems.

To advance the performance boundaries GNNs, a variety of enhancement strategies have been explored, which can be broadly categorized into three paradigms: architectural refinement, representation-level learning, and knowledge integration. Architectural improvements aim to strengthen message propagation by introducing deeper or residual structures, such as GCNII (Chen et al. 2020) and JKNet (Xu et al. 2018), or adaptive neighborhood sampling as in GraphSAINT (Zeng et al. 2020). Representation-level approaches focus on auxiliary learning objectives, including contrastive learning (e.g., GraphCL (You et al. 2020), BGRL (Thakoor et al. 2021)) and self-supervised pretraining (e.g., GPT-GNN (Hu et al. 2020b)), to improve embedding generalization. More recently, knowledge-based methods have attracted increasing attention, particularly those integrating external semantic signals from large language models (LLMs). For instance, LLM-GNN (Chen et al. 2024b) augments node features with textual context, while GraphToolformer (Zhang 2023) and GraphLLM (Chai et al. 2023) guide message passing or prediction using LLM-generated knowledge. These approaches show the potential of LLMs in enriching GNNs with high-level semantic priors, paving the way for a new direction in graph representation learning.

While existing studies have preliminarily demonstrated the feasibility of incorporating LLMs into GNNs to improve performance, the full potential of LLMs as high-quality

*Corresponding authors.

semantic supervision sources remains underexplored. Current approaches predominantly focus on extracting semantically aligned positive information to enhance representation, yet often overlook contrastive signals that, despite being semantically dissimilar, are more valuable for constructing inter-class decision boundaries. In principle, LLMs possess the capacity to generate multi-dimensional, task-relevant knowledge—including both positive signals that reinforce intra-class consistency and negative cues that highlight inter-class distinctions. However, effectively exploiting such rich semantics remains challenging. First, the quality of LLM outputs is highly dependent on prompt design, and generating informative yet complementary signals requires carefully contextualized guidance. Second, graph data commonly contain noise, ambiguity, or adversarial perturbations, which may trigger hallucinations in LLM outputs, undermining the reliability of knowledge transfer. Third, existing methods generally lack dynamic mechanisms to assess and filter generated signals. Although some works adopt distillation techniques to guide GNN learning, they still largely rely on static transfer of positive semantics, failing to systematically mine and integrate discriminative external knowledge (Hu et al. 2025; Pan et al. 2024). These challenges underscore the need for a robust and controllable knowledge transfer framework to fully unlock the supervisory potential of LLMs in graph representation learning.

To address above challenges, we propose DuoKD, a dual positive-negative semantic distillation framework that harnesses LLMs to provide fine-grained and robust supervision for GNNs. Specifically, we design personalized prompting strategies to elicit from LLMs both semantically similar positive signals and semantically dissimilar negative signals, which enhance the model's ability to capture intra-class consistency and inter-class distinction. In addition, we extract structural and semantic reasoning explanations corresponding to these signals, which serve as auxiliary knowledge to strengthen the informativeness and reliability of the supervision. To ensure the quality of transferred knowledge, we introduce a natural language-based evaluation mechanism to filter out low-confidence or hallucinated outputs, enabling the distillation of trustworthy signals. Finally, under a unified knowledge distillation framework, DuoKD adaptively integrates positive and negative knowledge to guide the GNN's training process, facilitating robust representation learning in the presence of noisy or ambiguous graph data. In summary, our core contributions are three-fold:

- We propose a dual positive-negative knowledge augmentation paradigm that systematically incorporates semantically similar positive signals and semantically dissimilar negative signals from LLMs, explicitly modeling intra-class consistency and inter-class distinctiveness to exploit the discriminative potential of external knowledge.
- We propose a unified knowledge distillation framework that dynamically transfers multi-dimensional signals from LLMs. It incorporates both opinion- and rationale-level supervision, while employing a language-based evaluation mechanism to filter low-confidence or noisy knowledge, thereby enabling robust GNN training.
- Extensive experiments on multiple benchmarks demonstrate that our method consistently outperforms existing approaches across various tasks, showing superior discriminative power and robustness, particularly under semantically ambiguous or structurally noisy scenarios.

Preliminaries

Text-Attributed Graphs. A Text-Attributed Graph (TAG) is formally defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X})$, where $\mathcal{V} = \{v_i\}_{i=1}^n$ represents the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes the edge set, and $\mathcal{X} = \{x_i\}_{i=1}^n$ consists of textual attributes for each node v_i. Here, x_i captures the natural language description of node v_i, enabling the integration of graph topology with rich semantic information. TAGs serve as foundational structures for applications like citation networks and knowledge graphs, where both relational dependencies and textual context are critical. However, modeling such graphs requires joint reasoning over structural dependencies and linguistic semantics, motivating the integration of GNNs and LLMs.

Graph Neural Networks. GNNs process graph-structured data through iterative message passing. For a node v, its representation at layer k is updated via:

$$\mathbf{h}_v^{(k)} = f_\theta \left(\mathbf{h}_v^{(k-1)}, \bigoplus_{u \in \mathcal{N}(v)} g_\phi \left(\mathbf{h}_u^{(k-1)}, \mathbf{h}_v^{(k-1)} \right) \right), \quad (1)$$

where $\mathbf{h}_v^{(k)} \in \mathbb{R}^d$ is the feature vector of node v at layer k, $\mathcal{N}(v)$ denotes its neighbors, g_ϕ aggregates neighbor features, f_θ updates node states, and $\bigoplus$ is an aggregation operator. While GNNs effectively capture multi-hop dependencies via message passing, they are limited in understanding the semantic complexity of textual attributes, highlighting the need for integration with LLMs.

Methodology

We commence with a high-level overview of our method, and subsequently detail each constituent module.

Overview

To fully leverage the semantic supervision capacity of LLMs while mitigating their unreliability and maximizing discriminative effectiveness in graph learning, we propose a dual knowledge distillation framework (DuoKD), structured around three core modules: (1) Positive-Negative Knowledge Extraction from LLMs, where large language models are guided to generate task-relevant opinions, rationales with both semantically aligned (positive) and contrasting (negative) perspectives; (2) Language-based Knowledge Evaluation, which utilizes natural language probing and counterfactual reasoning to assess the reliability of generated knowledge and filter out low-confidence or hallucinated signals; (3) Adaptive Dual Knowledge Distillation, which distills high-confidence knowledge into lightweight GNNs, encouraging them to align with informative positive guidance while learning from adversarially constructed negative supervision. This unified design effectively integrates the

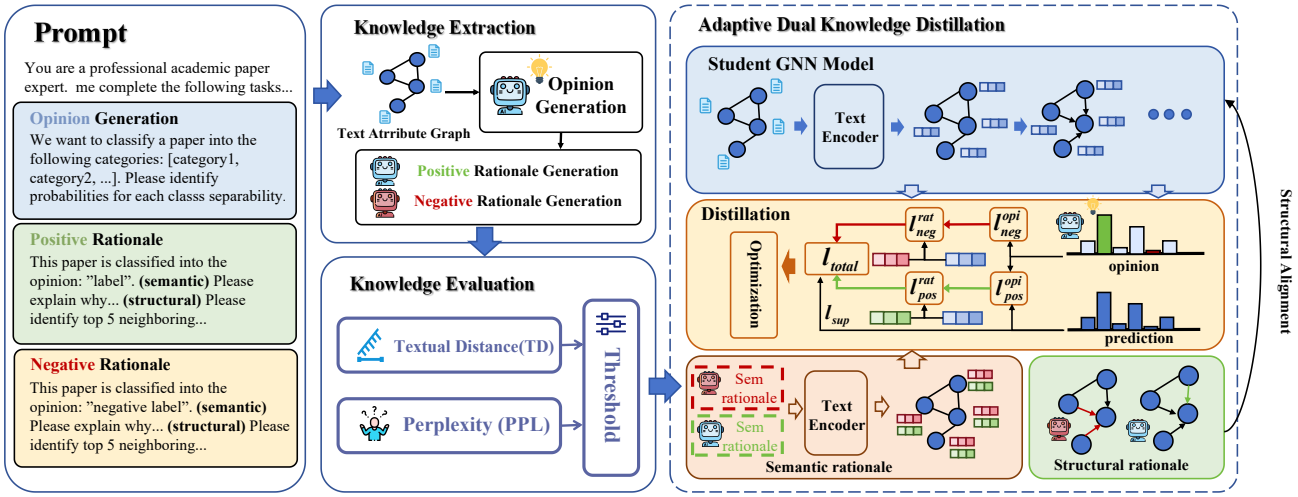

Figure 1: The whole architecture of our proposed DuoKD.

rich semantic priors of LLMs with the discriminative training of GNNs, enabling robust graph representation learning through dual positive-negative knowledge transfer.

Positive-Negative Knowledge Extraction

This module aims to extract opinion-level and rationale-level semantic knowledge from LLMs. To this end, we design personalized prompting strategies to guide the LLM in producing logits-like confidence scores over all predefined categories, while explicitly requiring comprehensive and fine-grained semantic perception across the entire label space. Based on the resulting outputs, we identify the most semantically aligned opinion as the positive supervision signal and the most semantically dissimilar category as the negative contrastive signal, along with their corresponding natural language rationales. These multi-dimensional signals are then transformed into discriminative and instructive external supervision to enhance the GNNs. To extract high-quality supervision signals from LLMs, we adopt a three-stage prompting-based framework that retrieves both predictive and explanatory knowledge for each node in a graph.

Step 1: Generation of Positive and Negative Opinions. To extract fine-grained semantic supervision, we leverage a zero-shot prompting strategy to query the LLM using both the node v_n's raw text attribute x_n and its neighborhood context $\{x_i\}_{i|v_i \in \mathcal{N}(v_n)}$. This produces a soft label vector $l_n \in \mathbb{R}^{|\mathcal{Y}|}$, representing the LLM's logits-like confidence distribution over predefined classes. Unlike traditional pseudo-labeling, our approach enables the LLM to assess semantic alignment and inter-class distinction under a graph-aware context, offering both the most semantically aligned class as the positive opinion and the most dissimilar one as the negative opinion. The process is formulated as:

$$l_n = \texttt{LLM}(x_n, \{x_i\}; \texttt{prompt}_{\text{label}}), \tag{2}$$

We then derive the positive and negative opinions based on the soft label distribution l_n. Specifically, the class with the highest confidence is selected as the positive label $y_n^+ = \arg\max(l_n)$, while the class with the lowest confidence (i.e., the minimum value in l_n) is selected as the negative label $y_n^- = \arg\min(l_n)$ to provide contrastive supervision:

$$y_n^+ = \arg\max_{y \in \mathcal{Y}} l_n[y], \tag{3}$$

$$y_n^- = \arg\min_{y \in \mathcal{Y}} l_n[y]. \tag{4}$$

The jointly extracted positive and negative opinions capture the LLM's semantic perception of the input and enables the GNN to simultaneously enhance intra-class consistency and inter-class discrimination.

Prompt Example: Generation of Opinions

Input: We want to classify a paper into the following categories: [category1, category2, ...]. Please identify logits-like probabilities for each class and give your final classification, ensuring that the distribution exhibits maximal inter-class separability. At the same time, the category with the lowest confidence is identified as negative label
The paper is: "...".
Its neighbors are: "..."
Response: {Probabilities: [0.1, 0.9, ...], Positive Opinion: 'category2', Negative Opinion:'category1'}

Step 2: Positive Rationale Extraction. Unlike traditional knowledge distillation methods that primarily rely on soft labels or logits from LLMs (Chen et al. 2024b; Chai et al. 2023), our framework introduces a richer form of supervision from LLMs, comprising two key components: the predicted opinion y_n^+, and a pair of rationales that provide semantic and structural explanations for this prediction. These rationales aim to bridge the LLM's natural language understanding and the GNN's symbolic graph-based reasoning.

- **Semantic rationale:** explains why the textual content supports the predicted opinion.

- **Structural rationale:** highlights how the graph structure (e.g., neighborhood information) reinforces the classification decision.

Prompt Example: Positive Rationale

Input: This paper is classified into the opinion: "label". **(semantic rationale)** Please explain why this classification makes sense. What characteristics or signals in the paper content might justify this label? **(structural rationale)** Please identify the top 5 neighboring papers that most strongly support this classification.
The paper is: "...".
Its neighbors are: "..."
Response: {semantic rationale: 'The classification of this paper into the category is justified due to its focus on the foundational....', structural rationale: ['node id',cofidence,...]

In this context, we treat LLMs not only as label teachers but also as knowledge supervisors, providing semantic and structural justifications that GNNs can align with during training. For each node v_i, the LLM produces a rationale R_i that reflects the internal decision-making process. We divide R_i into two components:

- The **semantic rationale** $r^{+}_{\text{sem},i}$, which is embedded using a language encoder (e.g., SBERT(Reimers and Gurevych 2019)) into a dense vector representation $\phi^{LLM}(r^{+}_{\text{sem},i})$, and serves as a target for aligning the GNN's latent representation via MSE loss;
- The **structural rationale** $r^{+}_{\text{struct},i}$, which is parsed into a set of relevance scores over the node's neighbors. These scores are normalized and used as confidence-aware edge weights that modulate message passing within GNN.

Prompt Example: Negative Rationale

Input: This paper is classified into the category: "negative label". **(semantic rationale)**Please explain why this classification makes sense. What characteristics or signals in the paper content might justify this label? **(structural rationale)**Please identify the top 5 neighboring papers that most strongly support this classification.
The paper is: "...".
Its neighbors are: "..."
Response: {semantic rationale: 'The classification of this paper into the category is justified due to its focus on the foundational....', structural rationale: ['node id',cofidence,...]

Step 3: Negative Rationale Extraction. To obtain negative supervision signals, we draw inspiration from the paradigm of counterfactual explanation generation (Feng et al. 2024), which encourages a model to reason about "what would need to change for a different decision to be made." In our case, we exploit this capacity of LLMs by deliberately injecting a negative label y_n^- into the prompt to construct a counterfactual scenario. Instead of asking the LLM to predict or justify the correct label, we pretend that the node should belong to y_n^- and ask the LLM to rationalize that false assumption.

This subtle manipulation leads the LLM to expose weaknesses or inconsistencies between the input node and the incorrect label, thus generating valuable negative rationales. The negative rationales $r^{-}_{\text{sem},n}$ and $r^{-}_{\text{struct},n}$ are defined analogously to their positive counterparts, but are generated by conditioning the LLM on a counterfactual label y_n^- to simulate incorrect reasoning paths.

After generating these rationales, we draw inspiration from negative learning (Kim et al. 2019, 2021), which encourages the student model to explicitly diverge from misleading supervision. Instead of mimicking these rationales, the GNN is penalized when its internal representations align with the LLM's explanation for the incorrect class. This forms the basis for our contrastive rationale loss .

Language-based Knowledge Evaluation

To ensure the reliability of natural language rationales used in distillation, we perform a language-level evaluation over both positive and negative semantic rationales r^{+}_{sem} and r^{-}_{sem}. Specifically, we adopt two complementary metrics from the CEval benchmark (Nguyen, Schlötterer, and Seifert 2024) to assess both the semantic consistency and linguistic fluency of generated rationales:

- **Textual Distance (TD):** We compute the average Levenshtein distance between each generated rationale x and its counterfactual reference x_{cf}, measuring their semantic and lexical deviation:

$$\text{TD}(x, x_{cf}) = \frac{1}{N}\sum_{i=1}^{N} \text{Levenshtein}(x^{(i)}, x_{cf}^{(i)}), \quad (5)$$

  where N is the batch size, and $\text{Levenshtein}(\cdot,\cdot)$ calculates the minimum token edits (insertions/deletions/substitutions) between text pairs.
- **Perplexity (PPL):** We employ a pretrained GPT-2 model to evaluate the fluency of each rationale x_{cf}. The perplexity is defined as:

$$\text{PPL}(x_{cf}) = \exp\left(-\frac{1}{T}\sum_{t=1}^{T}\log P_{\text{GPT-2}}(w_t \mid w_{<t})\right), \quad (6)$$

  where T is the number of tokens in x_{cf}, and $P_{\text{GPT-2}}$ denotes the GPT-2 model probability.

Rationales with high TD or abnormal PPL are discarded before knowledge transfer, ensuring that only high-quality and faithful supervision signals are distilled into the GNN training process. Then each rationale r is scored by a weighted sum of the two metrics:

$$\text{Score}_{\text{sem}}(r) = \alpha \cdot \text{TD}(r) + (1-\alpha)) \cdot \text{PPL}(r), \quad (7)$$

where α is the hyperparameter that balance textual deviation and fluency. A rationale is retained for training only if its score falls below a predefined threshold. This filtering mechanism reduces the impact of noisy generation and ensures that the GNN is guided by trustworthy semantic explanations. To apply this evaluation in practice, we introduce a binary gating function:

$$\mathbb{1}[\text{Score}_{\text{sem}}(r) < \tau], \quad (8)$$

where τ is a tunable threshold. This indicator function equals 1 if the rationale's score is acceptable and 0 otherwise. We use this gate to control whether a rationale contributes to the rationale-level alignment loss.

Adaptive Dual Knowledge Distillation

Our DuoKD framework leverages both positive and negative supervision signals extracted from LLMs to enhance the learning of GNNs on text-attributed graphs. The distillation operates on two complementary axes: opinion-level alignment and rationale-level alignment, each containing positive and negative components.

Opinion-Level Distillation. At the prediction level, the GNN is guided to mimic the soft pseudo-labels $\mathbf{z}^{(LLM)}$ produced by the LLM. Positive label loss pushes the student to match the teacher's output distribution:

$$\mathcal{L}_{pos}^{opi} = \mathrm{KL}\big(\sigma(\mathbf{z}^{(LLM)}/\tau) \parallel \sigma(\mathbf{z}^{(GNN)}/\tau)\big), \quad (9)$$

where τ is a temperature parameter and σ denotes the softmax function.

To avoid misleading the GNN with unreliable or incorrect supervision, we introduce a negative label loss that explicitly penalizes the student model when its predictions are overly aligned with the LLM's outputs for negative (low-confidence or incorrect) labels:

$$\mathcal{L}_{neg}^{opi} = \sum_{i \in \mathcal{I}_{neg}} \max\big(0, \cos(\mathbf{z}_i^{(GNN)}, \mathbf{z}_i^{(LLM)}) - m\big), \quad (10)$$

where $\mathcal{I}_{neg}$ indexes negative samples and m is a margin hyperparameter.

Rationale-Level Distillation. Beyond label predictions, we also distill explanatory knowledge by aligning the GNN's internal representations with LLM-generated rationales. These rationales come in two modalities: semantic (natural language explanations) and structural (neighbor importance scores).

Semantic rationales. $r_{\mathrm{sem},i}^{+}$ and $r_{\mathrm{sem},i}^{-}$ are embedded into vector representations $\phi^{LLM}(r)$ using a pretrained language encoder. To make the representation spaces compatible, the GNN node embeddings are first mapped through a learnable projection head P, producing $\tilde{\phi}^{GNN}(v_i) = P(\phi^{GNN}(v_i))$. These projected embeddings $\tilde{\phi}^{GNN}(v_i)$ are then trained to align with positive semantic rationales while being repelled from negative ones, gated by the quality score in Section 3.2:

$$\mathcal{L}_{pos}^{rat} = \mathbb{1}\big[\mathrm{Score}_{\mathrm{sem}}(r_i^{+}) < \tau\big] \cdot \mathrm{MSE}\big(\tilde{\phi}^{GNN}(v_i), \phi^{LLM}(r_i^{+})\big), \quad (11)$$

$$\mathcal{L}_{neg}^{rat} = \mathbb{1}\big[\mathrm{Score}_{\mathrm{sem}}(r_i^{-}) < \tau\big] \cdot \max\big(0, \cos(\tilde{\phi}^{GNN}(v_i), \phi^{LLM}(r_i^{-})) - m\big). \quad (12)$$

Here, $\mathbb{1}[\cdot]$ is an indicator gating function that filters out low-quality rationales based on a threshold τ.

Structural rationales. $r_{\mathrm{struct},i}^{+}$ produced by the LLM assign confidence scores to a node's neighbors, indicating their relevance for classification. We integrate these importance weights $\mathbf{w}_i \in \mathbb{R}^{|\mathcal{N}(v_i)|}$ into the GNN's message passing by modulating edge weights accordingly:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \mathbf{w}_{ij} \cdot \mathbf{W}\mathbf{h}_j^{(l)}\right), \quad (13)$$

where $\mathbf{W}$ is a learnable transformation matrix and σ is an activation function. This mechanism allows the GNN to preferentially aggregate information from semantically important neighbors as indicated by the LLM. Negative structural rationales $r_{\mathrm{struct},i}^{-}$ can be used to penalize the GNN's reliance on misleading neighbors by contrasting the GNN's learned attention weights with the inverted negative importance distribution. Accordingly, the positive and negative distillation losses can be defined as:

$$\mathcal{L}_{\mathrm{pos}} = \mathcal{L}_{pos}^{opi} + \mathcal{L}_{pos}^{rat}, \quad (14)$$

$$\mathcal{L}_{\mathrm{neg}} = \mathcal{L}_{neg}^{opi} + \mathcal{L}_{neg}^{rat}. \quad (15)$$

Overall Objective. The overall training objective combines both types of supervision in a unified distillation loss:

$$\mathcal{L}_{\mathrm{total}} = \lambda_1 \cdot \mathcal{L}_{\mathrm{pos}} + \lambda_2 \cdot \mathcal{L}_{\mathrm{neg}} + \lambda_3 \cdot \mathcal{L}_{\mathrm{sup}}, \quad (16)$$

where $\mathcal{L}_{sup}$ is the standard supervised loss, λ controls the importance of different supervision signals during training.

This dual framework enables the GNN to absorb rich, trustworthy semantic and structural knowledge from the LLM while explicitly avoiding misleading or noisy signals, leading to improved robustness and interpretability.

Experiments

Experimental Setup

Datasets. We evaluate our framework on three widely-used text-attributed graph benchmarks: **Cora**, **PubMed**, and **ogbn-arxiv**. **Cora** and **PubMed** (Sen et al. 2008) are citation networks where nodes represent scientific publications and edges denote citation links. Each node is associated with a document abstract and a single topic label. **ogbn-arxiv** is a large-scale academic graph from the Open Graph Benchmark (OGB) (Hu et al. 2020a), where nodes correspond to arXiv papers indexed by MAG and edges represent citation relations. Each node is labeled with one of 40 subject areas and is associated with a title and abstract. In accordance with established methodology, all datasets were partitioned into training, validation, and test subsets using a randomized 60-20-20 split ratio.

Implementation Details. Our implementation is based on PyTorch Geometric and Huggingface Transformers (MIT License). All experiments are conducted on a single NVIDIA RTX 3090 GPU (24GB VRAM). We use GPT-4o-mini as the frozen LLM teacher, with a maximum input length of 512 tokens for full-text prompts and 48 tokens for keyword-style prompts. For student models, we adopt 2-layer GNN with hidden dimension 256. We first query the LLM to extract all signals, then freeze them and train the student GNN with our proposed loss. GNNs are trained for 200 epochs with learning rate 0.01. Evaluation metrics are Accuracy and Macro-F1, averaged over 5 random seeds.

Method	Cora		PubMed		ogbn-arxiv	
	Acc	F1	Acc	F1	Acc	F1
GCN	86.53 ± 0.92	85.36 ± 1.01	86.15 ± 0.87	85.64 ± 0.82	71.74 ± 0.21	70.44 ± 0.33
GAT	86.10 ± 1.00	85.12 ± 1.03	86.20 ± 0.93	85.31 ± 0.88	73.66 ± 0.35	71.04 ± 0.47
GraphSAGE	86.92 ± 0.85	85.85 ± 0.79	87.12 ± 0.76	86.90 ± 0.70	73.85 ± 0.43	72.31 ± 0.50
GIN	86.60 ± 0.91	85.41 ± 0.88	86.40 ± 0.95	85.78 ± 0.84	71.62 ± 0.42	70.33 ± 0.47
FAGCN	88.32 ± 0.55	86.93 ± 0.61	88.64 ± 0.63	87.96 ± 0.58	74.21 ± 0.58	72.97 ± 0.62
GIANT (GCN)	87.74 ± 0.85	86.30 ± 0.78	88.32 ± 0.70	87.05 ± 0.69	75.61 ± 0.67	74.25 ± 0.65
GLEM (GCN)	88.71 ± 0.61	87.51 ± 0.65	88.74 ± 0.62	88.04 ± 0.57	76.20 ± 0.44	75.20 ± 0.47
GLEM (GAT)	89.12 ± 0.54	88.04 ± 0.60	89.01 ± 0.55	88.36 ± 0.48	76.42 ± 0.52	75.43 ± 0.50
GLEM (SAGE)	88.90 ± 0.47	87.92 ± 0.51	89.30 ± 0.58	88.62 ± 0.53	76.80 ± 0.41	75.91 ± 0.43
LinguGKD (GCN)	90.50 ± 0.42	88.90 ± 0.43	88.78 ± 0.54	88.66 ± 0.57	76.15 ± 0.51	76.92 ± 0.50
LinguGKD (GAT)	90.83 ± 0.46	89.21 ± 0.50	88.93 ± 0.47	87.90 ± 0.45	77.62 ± 0.40	77.21 ± 0.39
LinguGKD (SAGE)	90.91 ± 0.43	89.17 ± 0.44	88.10 ± 0.42	88.14 ± 0.38	77.05 ± 0.39	77.89 ± 0.41
Ours (GCN)	**91.91 ± 0.57**	**89.10 ± 0.63**	**90.93 ± 0.49**	**89.89 ± 0.55**	**78.84 ± 0.66**	**78.52 ± 0.61**
Ours (GAT)	**92.30 ± 0.51**	**89.60 ± 0.58**	**89.21 ± 0.42**	**89.04 ± 0.48**	**80.13 ± 0.59**	**78.81 ± 0.50**
Ours (SAGE)	**91.85 ± 0.63**	**88.92 ± 0.71**	**88.05 ± 0.44**	**88.78 ± 0.52**	**80.01 ± 0.47**	**78.66 ± 0.46**

Table 1: Node classification Accuracy and Macro-F1 (%) on three datasets. All values are reported as mean ± std over 5 runs.

Baselines. To evaluate the performance of our proposed framework, we compare it with a broad set of representative graph learning models. These baselines cover both classical and modern graph architectures, ranging from simple message-passing GNNs to models enhanced by pretrained language models. Specifically, we include **GCN** (Kipf and Welling 2017), **GAT**(Velickovic et al. 2018), **GraphSAGE** (Hamilton, Ying, and Leskovec 2017), **GIN** (Xu et al. 2019), and **FAGCN** (Bo et al. 2021) as foundational GNN baselines, which are widely adopted for node classification tasks. We further include LLM-augmented frameworks such as **GIANT** (Chien et al. 2023) and **GLEM** (Zhao et al. 2023) (combined with GCN, GAT, and SAGE backbones), which enhance GNNs with textual semantics via pretrained language models. Finally, we consider recent LLM-to-GNN distillation approaches, including **LinguGKD** (Hu et al. 2025)(combined with GCN, GAT, and SAGE backbones) .

Overall Performance

Table 1 presents the overall performance of all methods on the Cora, PubMed, and ogbn-arxiv datasets. We report both Accuracy and Macro-F1, averaged over five runs. Across all three datasets, our proposed dual distillation framework achieves consistent and significant improvements over classical GNNs, LLM-enhanced models, and recent LLM distillation baselines. For instance, on the Cora dataset, our best model (GAT student) reaches an accuracy of 92.30% and Macro-F1 of 89.60%, outperforming both the best classical GNN (FAGCN: 88.32/86.93) and advanced distillation methods such as LinguGKD (SAGE: 90.91/89.17). Similar trends are observed on PubMed and ogbn-arxiv, where our distilled GraphSAGE student reaches 88.05% accuracy on PubMed and 80.01% on ogbn-arxiv, showing strong generalization on both medium- and large-scale datasets.

Compared with LLM-enhanced methods such as GIANT and GLEM, our method offers substantial gains while avoiding the need for joint model training or fine-tuning large language models. Moreover, our framework remains architecture-agnostic, as evidenced by the consistent improvement across GCN, GAT, and GraphSAGE backbones.

These results validate the effectiveness of our distillation approach in transferring both opinion and rationale-level knowledge from LLMs into GNNs, enabling strong performance even in supervised settings with limited training data.

Robustness Analysis

To evaluate our framework's resilience, we conducted a robustness analysis against structural and attribute perturbations, using the non-targeted DICE attack (Waniek et al. 2018) and random text replacement, respectively. The results, detailed in Table 2, unequivocally demonstrate our model's superior stability across all datasets and attack intensities. For instance, under a 40% structural attack on Cora, our model maintains an accuracy of 84.21% $\pm$ 0.38, significantly outperforming all baselines. This resilience is evident under attribute perturbation, achieving 84.90% $\pm$ 0.34 accuracy on Pubmed with 40% corruption.

The model's exceptional robustness stems from its ability to synthesize multiple forms of knowledge, drawing heavily on the principles of negative learning. By integrating not only positive knowledge (positive opinion and supporting rationales) but also valuable negative knowledge (e.g., counter-evidence from negative opinion and rationales), our framework builds a more comprehensive and discriminative understanding. This negative knowledge explicitly teaches the model to recognize and reject misleading signals, thereby strengthening its decision boundaries. This makes the student model less reliant on any single information source and highly resilient to attacks that degrade either the graph's topology or the nodes' textual quality.

Ablation Study

We conducted an ablation study to examine the individual contributions of five key supervision signals in our distillation framework: positive knowledge, negative knowledge, semantic rationale, structural rationale, and the combination of semantic and structural guidance—to evaluate their individual contributions. As shown in Figure 2, we report

Dataset	Ptb Rate	Structural Perturbation				Attribute Perturbation			
		GCN	GIANT	GLEM	Ours	GCN	GIANT	GLEM	Ours
Cora	0%	86.53 ± 0.92	87.74 ± 0.85	88.78 ± 0.46	91.91 ± 0.57	86.53 ± 0.92	87.74 ± 0.85	88.78 ± 0.46	91.91 ± 0.57
	10%	81.38 ± 0.65	84.21 ± 0.53	85.92 ± 0.42	86.35 ± 0.34	82.80 ± 0.58	85.45 ± 0.48	87.25 ± 0.39	89.42 ± 0.33
	20%	78.38 ± 0.70	82.42 ± 0.58	84.63 ± 0.48	85.90 ± 0.36	78.60 ± 0.67	82.10 ± 0.53	85.15 ± 0.45	87.30 ± 0.37
	40%	71.62 ± 0.76	77.93 ± 0.64	80.78 ± 0.52	84.21 ± 0.38	72.30 ± 0.74	76.45 ± 0.60	80.50 ± 0.48	85.00 ± 0.35
Pubmed	0%	86.15 ± 0.87	88.32 ± 0.70	88.68 ± 0.45	90.93 ± 0.49	86.15 ± 0.87	88.32 ± 0.70	88.68 ± 0.45	90.93 ± 0.49
	10%	82.86 ± 0.63	85.60 ± 0.50	87.02 ± 0.43	88.12 ± 0.33	83.25 ± 0.56	86.15 ± 0.44	87.90 ± 0.38	88.65 ± 0.32
	20%	80.11 ± 0.68	83.53 ± 0.55	85.84 ± 0.47	87.75 ± 0.35	78.95 ± 0.64	83.20 ± 0.51	86.10 ± 0.43	86.95 ± 0.36
	40%	76.15 ± 0.73	80.92 ± 0.61	83.67 ± 0.50	86.24 ± 0.37	73.15 ± 0.72	78.85 ± 0.57	83.30 ± 0.46	84.90 ± 0.34
ogbn-arxiv	0%	71.74 ± 0.21	75.61 ± 0.67	76.12 ± 0.42	78.84 ± 0.66	71.74 ± 0.21	75.61 ± 0.67	76.12 ± 0.42	78.84 ± 0.66
	10%	64.23 ± 0.69	67.54 ± 0.53	69.90 ± 0.45	71.12 ± 0.32	67.50 ± 0.62	71.55 ± 0.50	74.00 ± 0.41	76.85 ± 0.30
	20%	62.35 ± 0.72	65.83 ± 0.56	67.74 ± 0.47	69.80 ± 0.34	63.20 ± 0.66	68.20 ± 0.52	71.60 ± 0.44	74.35 ± 0.30
	40%	56.44 ± 0.76	61.29 ± 0.61	64.88 ± 0.50	67.65 ± 0.36	57.90 ± 0.71	63.40 ± 0.59	67.80 ± 0.47	72.00 ± 0.33

Table 2: Robustness analysis under structural and attribute perturbations. All values are accuracy (%) reported as mean $\pm$ standard deviation over 5 runs.

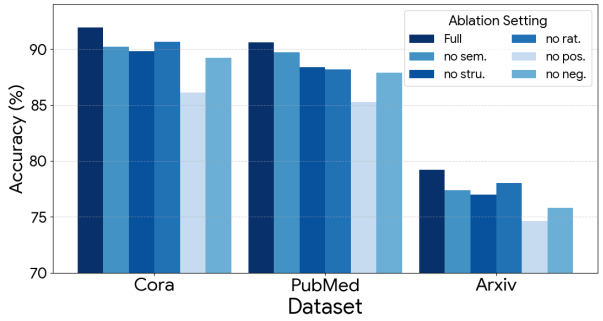

Figure 2: Performance comparisons of the new DuoKD with its three variantson three datasets.

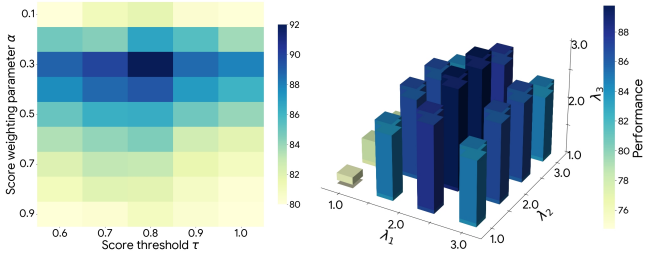

Figure 3: Parameter sensitivity analysis.

the classification accuracy across three datasets. The full model consistently achieves the best performance, indicating the effectiveness of integrating all three types of guidance. Notably, removing positive supervision (*no pos.*) leads to the most significant performance degradation, with accuracy dropping by 5.8%, 5.4%, and 4.6% on Cora, PubMed, and ogbn-arxiv respectively, highlighting its essential role in transferring task-specific knowledge.

Removing negative supervision (*no neg.*) causes moderate drops, especially on class-imbalanced datasets (e.g., ogbn-arxiv), confirming its role in reducing overconfidence on non-target classes. Excluding rationale supervision (*no rationale*, i.e., removing both semantic and structural guidance, and *no semantic* or *no structural* individually) leads to small but consistent drops, suggesting textual explanations are useful auxiliary signals. These findings validate our unified distillation design, where diverse teacher signals jointly improve student generalization.

Parameter Analysis

We conducted a parameter sensitivity analysis to evaluate the impact of key hyperparameters on our model's performance. The investigation focused on two main areas: the rationale evaluation parameters (α, τ) and the loss component weights (λ). For rationale filtering, performance is maximized when the weighting factor α is set between 0.2 and 0.4, and the filtering threshold τ is between 0.7 and 0.9, as shown in the heatmap. This highlights the necessity of a well-calibrated filter to ensure the quality of distilled knowledge. For the loss components, we analyzed the weights for positive knowledge (λ_1), negative knowledge (λ_2), and the supervised loss (λ_3). The 3D plot reveals that performance consistently improves as these weights increase, with optimal results achieved when all weights are set to higher values (e.g., 3.0–4.0).This finding highlights that emphasizing both positive guidance and negative counter-evidence is crucial for the student model's robustness.

Conclusion

In this paper, we introduce DuoKD, a novel dual knowledge distillation framework that leverages LLMs to enhance GNNs with discriminative and fine-grained supervision. By jointly modeling semantically aligned positive knowledge and semantically divergent negative knowledge, the proposed method explicitly captures intra-class consistency and inter-class separability. To ensure the reliability of transferred knowledge, we incorporate a language-based evaluation mechanism to assess and filter noisy or hallucinated signals. Subsequently, a unified distillation framework adaptively guides the GNN to align with high-confidence positive signals while learning to resist misleading negative cues. Extensive experiments on multiple benchmarks demonstrate the effectiveness and robustness of DuoKD, particularly in scenarios involving semantic ambiguity or structural noise.

Acknowledgments

This work was supported by National Key R&D Program of China (No.2023YFB2603902), the National Natural Science Foundation of China (No. 92370111, No. 62272340, No. 62422210 and No. 62276187), and the Postdoctoral Fellowship Program of CPSF under Grant No.GZC20251059. We would also like to express our sincere gratitude to Professor Zhanjie Song from the School of Mathematics, Tianjin University, for his invaluable guidance.

References

Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond Low-frequency Information in Graph Convolutional Networks. In *Proceedings of AAAI*, 3950–3957.

Cai, L.; Li, J.; Wang, J.; and Ji, S. 2021. Line graph neural networks for link prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9): 5103–5113.

Chai, Z.; Zhang, T.; Wu, L.; Han, K.; Hu, X.; Huang, X.; and Yang, Y. 2023. Graphllm: Boosting graph reasoning ability of large language model. *arXiv preprint arXiv:2310.05845*.

Chen, C.; Wu, Y.; Dai, Q.; Zhou, H.-Y.; Xu, M.; Yang, S.; Han, X.; and Yu, Y. 2024a. A survey on graph neural networks and graph transformers in computer vision: A task-oriented perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–20.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *Proceedings of ICML*, 1725–1735.

Chen, Z.; Mao, H.; Wen, H.; Han, H.; Jin, W.; Zhang, H.; Liu, H.; and Tang, J. 2024b. Label-free Node Classification on Graphs with Large Language Models (LLMs). In *Proceedings of SIGKDD*.

Chien, E.; Chang, W.; Hsieh, C.; Yu, H.; Zhang, J.; Milenkovic, O.; and Dhillon, I. S. 2023. Node Feature Extraction by Self-Supervised Multi-scale Neighborhood Prediction. In *Proceedings of ICLR*.

Dai, R.; Xu, S.; Gu, Q.; Ji, C.; and Liu, K. 2020. Hybrid spatio-temporal graph convolutional network: Improving traffic prediction with navigation data. In *Proceedings of SIGKDD*, 3074–3082.

Fan, W.; Ma, Y.; Li, Q.; He, Y.; Zhao, E.; Tang, J.; and Yin, D. 2019. Graph neural networks for social recommendation. In *Proceedings of the ACM Web Conference*, 417–426.

Feng, T.; Li, Y.; Chenglin, L.; Chen, H.; Yu, F.; and Zhang, Y. 2024. Teaching small language models reasoning through counterfactual distillation. In *Proceedings of EMNLP*, 5831–5842.

Gao, C.; Zheng, Y.; Li, N.; Li, Y.; Qin, Y.; Piao, J.; Quan, Y.; Chang, J.; Jin, D.; He, X.; et al. 2023. A survey of graph neural networks for recommender systems: Challenges, methods, and directions. *ACM Transactions on Recommender Systems*, 1(1): 1–51.

Hamilton, W. L.; Ying, Z.; and Leskovec, J. 2017. Inductive Representation Learning on Large Graphs. In *Proceedings of NeurIPS*, 1024–1034.

Hu, S.; Zou, G.; Yang, S.; Lin, S.; Gan, Y.; Zhang, B.; and Chen, Y. 2025. Large language model meets graph neural network in knowledge distillation. In *Proceedings of AAAI*, 17295–17304.

Hu, W.; Fey, M.; Zitnik, M.; Dong, Y.; Ren, H.; Liu, B.; Catasta, M.; and Leskovec, J. 2020a. Open graph benchmark: Datasets for machine learning on graphs. In *Proceedings of NeurIPS*, volume 33, 22118–22133.

Hu, Z.; Dong, Y.; Wang, K.; Chang, K.-W.; and Sun, Y. 2020b. Gpt-gnn: Generative pre-training of graph neural networks. In *Proceedings of SIGKDD*, 1857–1867.

Huo, C.; Jin, D.; Li, Y.; He, D.; Yang, Y.-B.; and Wu, L. 2023. T2-gnn: Graph neural networks for graphs with incomplete features and structure via teacher-student distillation. In *Proceedings of AAAI*, 4339–4346.

Jia, D.; Romić, I.; Shi, L.; Su, Q.; Liu, C.; Liu, J.; Holme, P.; Li, X.; and Wang, Z. 2025. Social networking agency and prosociality are inextricably linked in economic games. *Nature Human Behaviour*, 1–12.

Jin, D.; Huo, C.; Shi, J.; He, D.; Wei, J.; and Yu, P. S. 2025. Llgformer: Learnable long-range graph transformer for traffic flow prediction. In *Proceedings of the ACM on Web Conference*, 2860–2871.

Kim, Y.; Yim, J.; Yun, J.; and Kim, J. 2019. NLNL: Negative Learning for Noisy Labels. In *Proceedings of ICCV*, 101–110.

Kim, Y.; Yun, J.; Shon, H.; and Kim, J. 2021. Joint Negative and Positive Learning for Noisy Labels. In *Proceedings of CVPR*, 9442–9451.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *Proceedings of ICLR*.

Manipur, I.; Giordano, M.; Piccirillo, M.; Parashuraman, S.; and Maddalena, L. 2021. Community detection in protein-protein interaction networks and applications. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20(1): 217–237.

Nguyen, V. B.; Schlötterer, J.; and Seifert, C. 2024. CEval: A Benchmark for Evaluating Counterfactual Text Generation. *arXiv preprint arXiv:2404.17475*.

Pan, B.; Zhang, Z.; Zhang, Y.; Hu, Y.; and Zhao, L. 2024. Distilling large language models for text-attributed graph learning. In *Proceedings of CIKM*, 1836–1845.

Reimers, N.; and Gurevych, I. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Galligher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI magazine*, 29(3): 93–93.

Thakoor, S.; Tallec, C.; Azar, M. G.; Azabou, M.; Dyer, E. L.; Munos, R.; Veličković, P.; and Valko, M. 2021. Large-scale representation learning on graphs via bootstrapping. *arXiv preprint arXiv:2102.06514*.

Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *Proceedings of ICLR*.

Wang, Z.; Jusup, M.; Guo, H.; Shi, L.; Geček, S.; Anand, M.; Perc, M.; Bauch, C. T.; Kurths, J.; Boccaletti, S.; et al. 2020. Communicating sentiment and outlook reverses inaction against collective risks. *Proceedings of the National Academy of Sciences*, 117(30): 17650–17655.

Wang, Z.; Jusup, M.; Shi, L.; Lee, J.-H.; Iwasa, Y.; and Boccaletti, S. 2018. Exploiting a cognitive bias promotes cooperation in social dilemma experiments. *Nature communications*, 9(1): 2954.

Wang, Z.; Jusup, M.; Wang, R.-W.; Shi, L.; Iwasa, Y.; Moreno, Y.; and Kurths, J. 2017. Onymity promotes cooperation in social dilemma experiments. *Science advances*, 3(3): e1601444.

Waniek, M.; Michalak, T. P.; Wooldridge, M. J.; and Rahwan, T. 2018. Hiding individuals and communities in a social network. *Nature Human Behaviour*, 2(2): 139–147.

Wu, L.; Chen, Y.; Shen, K.; Guo, X.; Gao, H.; Li, S.; Pei, J.; Long, B.; et al. 2023. Graph neural networks for natural language processing: A survey. *Foundations and Trends in Machine Learning*, 16(2): 119–328.

Wu, S.; Sun, F.; Zhang, W.; Xie, X.; and Cui, B. 2022. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5): 1–37.

Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; and Philip, S. Y. 2020. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1): 4–24.

Xiao, S.; Wang, S.; Dai, Y.; and Guo, W. 2022. Graph neural networks in node classification: survey and evaluation. *Machine Vision and Applications*, 33(1): 4.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *Proceedings of ICLR*.

Xu, K.; Li, C.; Tian, Y.; Sonobe, T.; Kawarabayashi, K.-i.; and Jegelka, S. 2018. Representation learning on graphs with jumping knowledge networks. In *Proceedings of ICML*, 5453–5462.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph contrastive learning with augmentations. *Proceedings of NeurIPS*, 33: 5812–5823.

Zeng, H.; Zhou, H.; Srivastava, A.; Kannan, R.; and Prasanna, V. 2020. GraphSAINT: Graph Sampling Based Inductive Learning Method. In *Proceedings of ICLR*.

Zhang, J. 2023. Graph-toolformer: To empower llms with graph reasoning ability via prompt augmented by chatgpt. *arXiv preprint arXiv:2304.11116*.

Zhang, M.; and Chen, Y. 2018. Link prediction based on graph neural networks. *Proceedings of NeurIPS*, 5171–5181.

Zhang, X.-M.; Liang, L.; Liu, L.; and Tang, M.-J. 2021. Graph neural networks and their current applications in bioinformatics. *Frontiers in Genetics*, 12: 690049.

Zhao, J.; Qu, M.; Li, C.; Yan, H.; Liu, Q.; Li, R.; Xie, X.; and Tang, J. 2023. Learning on Large-scale Text-attributed Graphs via Variational Inference. In *Proceedings of ICLR*.

Zhou, J.; Cui, G.; Hu, S.; Zhang, Z.; Yang, C.; Liu, Z.; Wang, L.; Li, C.; and Sun, M. 2020. Graph neural networks: A review of methods and applications. *AI Open*, 1: 57–81.