Let Relations Speak: An End-to-End LLM-GNN Soft Prompt Framework for Fraud Detection

Zhixing Zuo[1], Huilin He[1], Jiasheng Wu[1], Dawei Cheng[1*]
[1]School of Computer Science and Technology, Tongji University
{2352469, huilin3}@tongji.edu.cn, wujiasheng@alumni.tongji.edu.cn
*** Correspondence:** dcheng@tongji.edu.cn

Abstract

In recent years, Large Language Models (LLMs) have shown great capability in processing graph tasks such as fraud detection. However, most existing methods rely heavily on rich text attributes, which poses difficulties for this domain due to the lack of textual data. Although some pioneering methods attempt to overcome it, their textualization of graph structures via hard prompts easily leads to feature distortion. Additionally, fraud detection often exhibits multi-relational complexity, where current methods struggle to capture this deep semantic information. To address these challenges, we propose LLM-GNN Soft Prompt Framework (LGSPF). Specifically, LGSPF bridges the graph structure and semantic space using soft prompt to eliminate reliance on text. We further introduce a parallel Graph Neural Network (GNN) encoder to translate multi-relational topologies into graph tokens for fine-grained LLM fraud comprehension. Through end-to-end optimization, LGSPF enhances deep semantic alignment between LLM and GNN. Experiments across diverse fraud detection benchmarks demonstrate our method achieves state-of-the-art performance. Moreover, we further validate the contribution of LGSPF on enhancing the semantic interpretability of fraud behaviors.

1 Introduction

Fraud detection has been widely deployed to safeguard diverse real-world systems, including financial networks (Huang et al., 2022), online review platforms (Wang et al., 2019), cybersecurity infrastructures (Lo et al., 2022), and social networks (Dou et al., 2020b). Given the nature of fraud activities, representing detection tasks as graphs has become a mainstream paradigm to capture suspicious interaction patterns (Ma et al., 2021). Recently, Large Language Models (LLMs) have exhibited remarkable reasoning and generalization abilities,

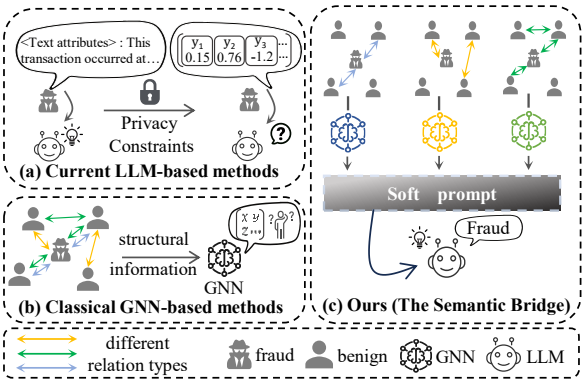

Figure 1: Comparison of different paradigms in Graph Fraud Detection: (a) Current LLM-based methods may fail in Weak-TAG scenarios. (b) Classical GNN-based methods lack interpretability on multi-relation graphs. (c) Our method seeks to bridge the semantic gap between GNNs and LLMs through soft prompt.

driving significant progress in various graph tasks (Li et al., 2024; Huang et al., 2024). However, deploying LLMs for fraud detection presents a unique and critical challenge: most existing LLM-based methods rely heavily on rich text attributes (Wang et al., 2025a), while real-world fraud graphs typically contain purely numerical features due to strict privacy and security constraints (Li et al., 2026a; Wang et al., 2025b). This leads to a Weak Text-Attributed Graph (Weak-TAG) scenario, where directly transferring existing LLM-based methods to such environment remains nontrivial.

Notably, some pioneering methods have attempted to reduce the reliance on textual attributes, among which graph textualization via hard prompts serves as a representative approach (Wu et al., 2025; Ye et al., 2024). By linearizing graph structures into natural language prompts, these methods offer a flexible and training-free solution that can be readily applied to various graph tasks. However, they suffer from structural-semantic misalignment. The graph topology, once flattened into natural language descriptions, loses its inherent relational se-

mantics, leading to feature distortion. (Zhang et al., 2026; Xu et al., 2026). To bridge this gap, recent soft prompt (Li and Liang, 2021; Lester et al., 2021) approaches, such as LLaGA (Chen et al., 2024) and GraphTranslator (Zhang et al., 2024), offer a promising solution by converting graph structures into LLM-compatible tokens.

However, directly applying soft prompt to encode graph into LLM for fraud detection faces challenges. In practice, fraudulent activities are rarely isolated; rather, they are highly dependent on complex, multi-relational interactions (Hooi et al., 2016). To address this complexity, Graph Neural Networks (GNNs) have emerged as powerful tools that aggregate neighborhood information to capture structural patterns (Han et al., 2025; Haghighi et al., 2024). Nevertheless, conventional GNNs essentially reduce rich relational information into static topological constraints or simplistic linear transformations, lacking interpretability for different edge types. To overcome this, coupling multi-relational GNNs with LLMs via soft prompts appears to be a promising direction. Unfortunately, many existing LLM-GNN frameworks rely on lightweight linear projector (Tang et al., 2024a; Cao et al., 2025) or frozen LLMs (Zhang et al., 2024), which disconnect structural encoding from semantic reasoning — the LLM's deep semantic feedback cannot be propagated to refine graph feature learning. For fraud detection, this disconnection cripples the model's ability to exploit structural semantics for identifying suspicious behaviors.

To summarize, Figure 1 illustrates the problem that most existing LLM-based and GNN-based methods are facing in fraud detection. It motivates us to explore the possibility in narrowing the gap between structural information and semantic reasoning, where GNNs are naturally fit in extracting topological features from multiple relations and LLMs specialize in deep semantic reasoning. We propose **LGSPF** (**L**LM-**G**NN **S**oft **P**rompt **F**ramework for Fraud Detection). In particular, the operation pipeline of LGSPF progresses through three highly correlated stages. First, for multi-relational graph feature encoding, we implement a parallel GNN encoder to capture structural representations under multiple relations and project them into the embedding space of LLMs. Subsequently, during soft prompt and semantic mapping, we introduce multiple special tokens as interfaces and map them with simple natural-language descriptions to help LLMs realize the semantic difference between relations. Finally, we design the training procedure as an end-to-end optimization, realizing a deep-level alignment of structure and semantics. Our contributions are summarized as follows:

- We propose LGSPF, an end-to-end LLM-GNN soft prompt framework for Fraud Detection, which bridges the gap between multi-relational graph structure and semantic space of LLMs under Weak-TAG scenarios.
- We introduce a parallel GNN encoder for feature extraction under a multi-relational graph, a soft prompt mechanism to seamlessly bridge graph structures with LLM reasoning, and an end-to-end training paradigm for deep structural-semantic alignment.
- Experiments conducted on three fraud detection datasets verify the effectiveness of our proposed framework. Furthermore, the results demonstrate that LGSPF delivers outstanding performance over several benchmarks and provides a new discovery in advancing graph fraud detection from black-box prediction to behavior recognition.

2 Related Work

2.1 LLMs for Graph Tasks

With the impressive abilities in generalization and reasoning, Large Language Models (LLMs) have been implemented to diverse graph tasks (Li et al., 2024). Recent advances can be categorized into LLM-as-enhancer and LLM-as-predictor. The former paradigm, including works like GALM (Xie et al., 2023) and LLMRec (Wei et al., 2024), focus on utilizing LLMs to process the text information and enhance the feature quality. LLM-as-predictor methods treats LLMs as predictors for diverse graph-related tasks such as classifications and reasonings. For instance, GraphGPT (Tang et al., 2024a) and GraphTranslator (Zhang et al., 2024) uses dual-stage instruction tuning to align LLMs with graph structural information, meanwhile HiGPT (Tang et al., 2024b) and DGP (Li et al., 2026b) implements instruction tuning to heterogeneous graphs. However, most of existing methods remain shackled by text attributes and lack an end-to-end tuning for deeply aligning structures and semantics, which our work addresses.

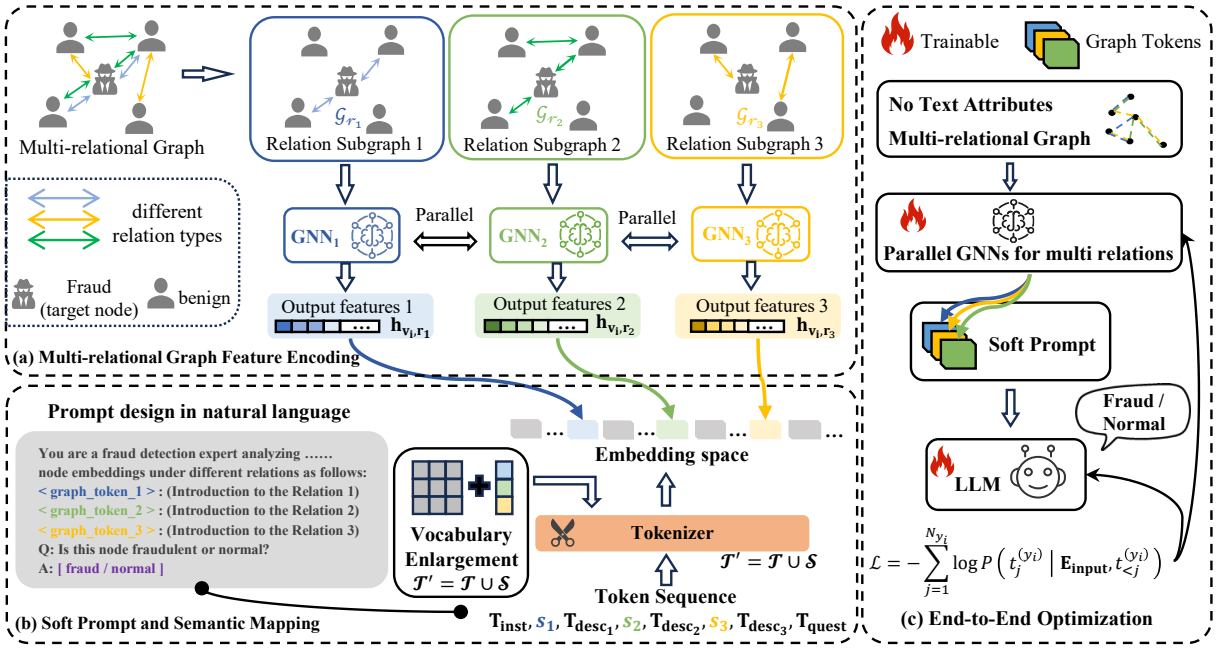

Figure 2: The overall architecture of our proposed LGSPF framework, which seamlessly aligns multi-relational graph structures with the large language model via an end-to-end soft prompting pipeline.

2.2 Fraud Detection on Graph

The topological structure of the graph provides a powerful discriminative basis for identifying fraudsters. Since the fraudulent activities exhibit complex interactive behaviors, researchers tend to construct multi-relation graphs with heterogeneous edges in distinct semantic types (Wu et al., 2020; Liu et al., 2018). To handle this challenge, diverse Graph Neural Network (GNN) based methods have emerged, leveraging community enhancement (Han et al., 2025), attention mechanisms (Haghighi et al., 2024), contrastive and reinforcement learning (Wang et al., 2023; Dou et al., 2020a). In contrast, recent works such as FLAG (Yang et al., 2025) and MLED (Huang et al., 2025) conduct LLMs to handle graph fraud detection, but they struggle to dynamically reason over the complex multi-relational topologies intrinsic to fraudulent behaviors. Differently, our LGSPF introduce soft prompt to bridge the GNNs and LLMs, successfully aligning the topological information in multi-relational graphs and the semantics reasoning, which significantly improve the performance and interpretability on fraud detection.

3 Methodology

Figure 2 illustrates the overall architecture of our proposed LGSPF. The pipeline is structured into three sections. Initially, for multi-relational graph feature encoding, we implement a parallel GNN encoder to capture relation-specific topological representations from independent subgraphs, projecting them to the embedding dimensions of the LLMs. Subsequently, during soft prompt and semantic mapping, we introduce special tokens to a unified prompt design, effectively bypassing flattened textualization. Finally, the framework employs an end-to-end optimization paradigm, where the GNN and LLM is jointly optimized to align the structural information and semantic reasoning.

3.1 Multi-relational Graph Feature Encoding

A graph with multi relations is defined as $\mathcal{G}_{r_i} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, where $\mathcal{V} = \{v_1, \ldots, v_n\}$ is the set of $n = |\mathcal{V}|$ nodes, $\mathcal{R} = \{r_1, \ldots, r_m\}$ is the set of m relations. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ denotes the set of edges. For a specific relation $r_i \in \mathcal{R}$, we define its corresponding subgraph as $\mathcal{G}_{r_i} = (\mathcal{V}, \mathcal{E}_{r_i})$, where $\mathcal{E}_{r_i} = \{(u, r_i, v) \mid (u, r_i, v) \in \mathcal{E}\}$. Consequently, we separate the original multi-relational graph into m homogeneous subgraphs, which satisfy:

$$\mathcal{E} = \bigcup_{i=1}^{m} \mathcal{E}_{r_i} \tag{1}$$

$$\mathcal{E}_{r_i} \cap \mathcal{E}_{r_j} = \emptyset, \forall i \neq j \in [1, m] \tag{2}$$

In the scenario of Weak-TAG, node v_i has a numeric feature vector $\mathbf{x}_i \in \mathbb{R}^d$ where d is the feature dimension. For the binary graph fraud detection dataset, the node v_i has label $y_i \in \{0, 1\}$, where

$y_i = 1$ indicates a fraudulent node and $y_i = 0$ indicates a normal node. In the semi-supervised setting, the overall node set $\mathcal{V}$ is partitioned into labeled $\mathcal{V}_{\mathrm{L}}$ and unlabeled $\mathcal{V}_{\mathrm{U}}$ subsets. While model optimization and evaluation are strictly confined to $\mathcal{V}_{\mathrm{L}}$, the unlabeled nodes in $\mathcal{V}_{\mathrm{U}}$ are preserved to provide fundamental topological information during the GNN message-passing phase.

Given a node $v_i \in \mathcal{V}$ as the detection target, our primary objective is to capture its diverse topological patterns under different relational contexts. To achieve this, we introduce a parallel GNN encoder. Specifically, we deploy an independent GNN branch for each relational subgraph $\mathcal{G}_{r_j}$ where $j \in [1, m]$. Notably, the weights are not shareable among different branches, which means that the distinct structural semantics inherent to each relation could be extracted respectively. This multi-view decoupling essentially provides fine-grained, relation-specific evidence for the subsequent cognitive reasoning using LLMs. Formally, the parallel encoding process is abstracted as:

$$\begin{aligned}\mathbf{h}_{v_i,r_j} =& \mathrm{GNN}_{r_j}(\mathbf{x}_i, \mathcal{G}_{r_j}; \mathbf{\Theta}_{r_j}) \\ & \in \mathbb{R}^{d_{\mathrm{emb}}}, \quad \forall j \in [1, m]\end{aligned} \tag{3}$$

where $\mathbf{\Theta}_{r_j}$ denotes the learnable parameters specific to relation r_j. Furthermore, in order to smoothly align with the latent embedding space of LLMs, we project the output dimension of all the GNN branches to match d_{emb}, facilitating seamless soft prompt integration.

It should be emphasized that our parallel architecture can adapt to various mainstream neural network-based graph structural encoders. In this work, we instantiate GNN_{r_j} using the highly recognized GraphSAGE (Hamilton et al., 2017) due to its outstanding performance in sparse graph situations. However, distinct from its original mini-batch sampling design, we perform full-batch neighborhood aggregation. This design choice guarantees deterministic message passing and prevents the structural information loss typically caused by random neighborhood sampling. For the l-th layer of the encoder under relation r_j, the representation of node v_i is updated through a two-stage paradigm:

$$\begin{aligned}\mathbf{h}^{(l)}_{\mathcal{N}_{r_j}(v_i)} = \mathrm{AGGREGATE}^{(l)}\Big(\\ \big\{\mathbf{h}^{(l-1)}_{u,r_j}, \forall u \in \mathcal{N}_{r_j}(v_i)\big\}\Big)\end{aligned} \tag{4}$$

$$\mathbf{h}^{(l)}_{v_i,r_j} = \sigma\Big(\mathbf{W}^{(l)}_{r_j} \cdot \big[\mathbf{h}^{(l-1)}_{v_i,r_j} \,\|\, \mathbf{h}^{(l)}_{\mathcal{N}_{r_j}(v_i)}\big]\Big) \tag{5}$$

Soft Prompt Design in LGSPF

$\mathbf{T}_{\mathrm{inst}}$: You are a fraud detection expert analyzing user behavior on a multi-relational graph dataset. We inject relation-specific structural representations into the hybrid template as follows:

<|s_1|>: $\mathbf{T}_{\mathrm{desc}_1}$ (Introduction to the Relation 1).
<|s_2|>: $\mathbf{T}_{\mathrm{desc}_2}$ (Introduction to the Relation 2).
<|s_3|>: $\mathbf{T}_{\mathrm{desc}_3}$ (Introduction to the Relation 3).

$\mathbf{T}_{\mathrm{quest}}$: `Q`: Is this node fraudulent or normal? You must analyze the multi-relational soft tokens and respond with one of these two words: 'fraud' or 'normal'. `A`:

Target Generation: `fraud` | `normal`

Table 1: The formal blueprint of the hybrid natural language template designed in LGSPF. The explicitly highlighted target space indicates the expected completion during the instruction tuning phase.

where $\mathcal{N}_{r_j}(v_i)$ represents the 1-hop neighbors of node v_i in subgraph $\mathcal{G}_{r_j}$. The $\mathrm{AGGREGATE}^{(l)}(\cdot)$ function is implemented as MEAN pooling, which computes the element-wise average of the neighbors' representations. $\sigma(\cdot)$ denotes the Exponential Linear Unit (ELU) activation function, and $\|$ represents the concatenation operation. $\mathbf{W}^{(l)}_{r_j}$ is the layer-specific transformation matrix.

The initial node representation is set to the input features $\mathbf{h}^{(0)}_{v_i,r_j} = \mathbf{x}_i$. We conduct K layers of aggregation (i.e., aggregating K-hop neighborhood features) to obtain the final outputs. Importantly, the learnable parameter set $\mathbf{\Theta}_{r_j}$ introduced in Eq. 3 is defined as below:

$$\mathbf{\Theta}_{r_j} = \left\{\mathbf{W}^{(l)}_{r_j}\right\}_{l=1}^{K}. \tag{6}$$

Ultimately, for each target node v_i, we extract a set of m relation-specific structural embeddings $\mathcal{H}_{v_i} = \{\mathbf{h}^{(K)}_{v_i,r_1}, \ldots, \mathbf{h}^{(K)}_{v_i,r_m}\}$.

3.2 Soft Prompt and Semantic Mapping

In natural language processing, soft prompt (Lester et al., 2021) is considered as a set of learnable continuous vectors, which does not match exact tokens in the vocabulary. Inspired by this, we treat the topological representations extracted by GNNs as conditional soft prompt. This design enables an alignment of the graph's structural characteristics with the latent space of the LLMs, preventing the loss of high-dimensional continuous information.

As explicitly exemplified in Table 1, we denote the vocabulary of the LLM as $\mathcal{T} = \{t_1, \ldots, t_{|\mathcal{T}|}\}$.

The token embedding matrix is defined as $\mathbf{W}_e \in \mathbb{R}^{|\mathcal{T}| \times d_{\text{emb}}}$, where d_{emb} is the embedding dimension. Specifically, given a discrete input token $t_i \in \mathcal{T}$ (where $i \in \{1, \ldots, |\mathcal{T}|\}$), the embedding layer applies a mapping function $f_{\text{emb}}(\cdot)$ to extract its corresponding continuous representation by retrieving the i-th row of the embedding matrix, denoted as

$$\mathbf{e}_{t_i} = f_{\text{emb}}(t_i) = \mathbf{W}_e[i, :] \in \mathbb{R}^{d_{\text{emb}}}. \tag{7}$$

In order to help LLMs distinguish the semantic difference between relations, we construct a hybrid prompt template $\mathcal{P}_{\text{temp}}$, which consists of natural language instructions and special tokens in an alternating manner :

$$\begin{aligned} \mathcal{P}_{\text{temp}} = [&\mathbf{T}_{\text{inst}}, s_1, \mathbf{T}_{\text{desc}_1}, \ldots, \\ &s_m, \mathbf{T}_{\text{desc}_m}, \mathbf{T}_{\text{quest}}] \end{aligned} \tag{8}$$

where the $\mathbf{T}_{\text{inst}}$ is defined as the token sequence of the global task instructions while the $\mathbf{T}_{\text{quest}}$ denotes the instructions of classification tasks. We realize the semantic mapping through the pairing of special token $s_j \in \mathcal{S}$ which represents the topological features under subgraphs $\mathcal{G}_{r_j}$ and the descriptions $\mathbf{T}_{\text{desc}_j}$ of relations r_j in natural language.

To accommodate the special tokens from s_1 to s_m, we enlarge the original vocabulary $\mathcal{T}$ to an augmented token space $\mathcal{T}' = \mathcal{T} \cup \mathcal{S}$ where $\mathcal{S} = \{s_1, \ldots, s_m\}$. Consequently, the entire hybrid sequence $\mathcal{P}_{\text{temp}}$ is first mapped into an initial embedding matrix $\mathbf{E}_{\text{temp}}$ via the pre-trained embedding layer $f_{\text{emb}}(\cdot)$:

$$\mathbf{E}_{\text{temp}} = f_{\text{emb}}(\mathcal{P}_{\text{temp}}) \in \mathbb{R}^{L \times d_{\text{emb}}} \tag{9}$$

where L denotes the total counts of token nums in the conditional soft prompt sequence.

Finally, to dynamically incorporate structural features of the target node under multiple relations, we perform a structure-aware injection, replacing the initial embeddings of the special tokens in $\mathbf{E}_{\text{temp}}$ with the relation-specific structural embeddings $\mathbf{h}_{v_i, r_j}$ extracted from the parallel GNN encoders. Let $\text{idx}(s_j)$ be the absolute position index of the special tokens s_j within the sequence $\mathcal{P}_{\text{temp}}$. The final input embedding matrix $\mathbf{E}_{\text{input}}$ is constructed as follows:

$$\mathbf{E}_{\text{input}}[k, :] = \begin{cases} \mathbf{h}_{v_i, r_j}, & \text{if } k = \text{idx}(s_j), \\ & \forall j \in [1, m] \\ \mathbf{E}_{\text{temp}}[k, :], & \text{otherwise} \end{cases} \tag{10}$$

Through this explicit cross-modal transition, the discrete special tokens are dynamically activated by continuous structural tensors, guiding the LLM to perform cognitive reasoning over complex multi-relational topologies.

3.3 End-to-End Optimization

We reframe the graph fraud detection task as a conditional text generation problem under a standard Causal Language Modeling paradigm. Specifically, we establish a deterministic mapping between the binary label space and the target text sequences:

$$\mathbf{Y}^{(y_i)} = \begin{cases} \text{“fraud”}, & \text{if } y_i = 1 \\ \text{“normal”}, & \text{if } y_i = 0 \end{cases} \tag{11}$$

To account for tokenizer variability across different LLM backbones, each target sequence is represented as a list of discrete tokens, i.e., $\mathbf{Y}^{(k)} = \{t_1^{(k)}, \ldots, t_{N_k}^{(k)}\}$ where $k \in \{0, 1\}$. The conditional generation probability for each candidate sequence $\mathbf{Y}^{(k)}$ is computed via the chain rule:

$$P(\mathbf{Y}^{(k)} \mid \mathbf{E}_{\text{input}}) = \prod_{j=1}^{N_k} P(t_j^{(k)} \mid \mathbf{E}_{\text{input}}, t_{<j}^{(k)}) \tag{12}$$

During the inference phase, the model computes the sequence-level log-likelihood for both candidate answers and generates the final predicted label:

$$\hat{y}_i = \arg\max_{k \in \{0,1\}} \log P(\mathbf{Y}^{(k)} \mid \mathbf{E}_{\text{input}}) \tag{13}$$

Our training goal is to minimize the discrepancy between the LLM's generated output and the ground-truth label text. We mask the labels for all prompt tokens, ensuring the loss is only computed on the answer tokens as follows:

$$\mathcal{L} = -\sum_{j=1}^{N_{y_i}} \log P(t_j^{(y_i)} \mid \mathbf{E}_{\text{input}}, t_{<j}^{(y_i)}) \tag{14}$$

To reduce computational costs during training, we apply Low-Rank Adaptation (Hu et al., 2022) exclusively to the attention layers of the LLM. Formally, the complete set of trainable parameters Ψ in LGSPF is formulated as:

$$\Psi = \{\mathbf{\Theta}_{r_j}\}_{j=1}^{m} \cup \{\mathbf{A}_l, \mathbf{B}_l\}_{l \in \mathcal{L}_{\text{LoRA}}} \tag{15}$$

where $\{\mathbf{A}_l, \mathbf{B}_l\}$ are the low-rank adapters in the l-th Transformer layer. Such an end-to-end optimization allows the LLM's semantic feedback to

Dataset	Nodes	Fraud	Labeled	Relation	Edges
Amazon	11,944	6.87%	72.33%	U-P-U	175,608
				U-S-U	3,566,479
				U-V-U	1,036,737
YelpChi	45,954	14.53%	100%	R-T-R	573,616
				R-U-R	49,315
				R-S-R	3,402,743
S-FFSD	77,881	6.75%	38.06%	Source	120,224
				Target	229,170
				Location	232,275
				Type	232,721

Table 2: Key properties of the evaluated datasets. To emphasize the Weak-TAG scenario, the node features in all three datasets are numeric.

directly supervise the structural feature extraction. Consequently, the GNN encoders can adaptively calibrate their topological representations to better align with the LLM's cognitive space, leading to a deeper structural-semantic convergence.

4 Experiments

4.1 Experimental Setup

Datasets To evaluate the effectiveness of our proposed LGSPF, we conducted experiments on three real-world public fraud detection datasets. The Amazon dataset (McAuley and Leskovec, 2013) contains a selection of reviews on products under the Musical Instruments category. The YelpChi dataset (Rayana and Akoglu, 2015) includes hotel and restaurant reviews on the Yelp platform. The S-FFSD is a simulated graph dataset from a financial fraud semi-supervised situation (Xiang et al., 2023). For the Amazon and YelpChi dataset, we process them into multi-relational graph following the same steps in Dou et al. (2020a). The key properties of the datasets are presented in Table 2. Detailed descriptions to the experimental datasets are provided in Appendix A.

Baselines We compared our LGSPF with a wide range of competitive models, which can be divided into two categories: (i) Classical GNNs, including GCN (Kipf and Welling, 2017), GAT (Veličković et al., 2018), GraphSAGE (Hamilton et al., 2017),CARE-GNN (Dou et al., 2020a), PC-GNN (Liu et al., 2021), BWGNN (Tang et al., 2022), GTAN (Xiang et al., 2023) and RGTAN (Xiang et al., 2025). We implemented these models using official code; (ii) Text-flattened LLMs, which directly serialize numerical features of the target node and its neighbor into discrete textual sequences, including Zero-shot LLM relying purely on pre-trained models and Instruct LLM fine-tuned via LoRA. Detailed prompt designs for them could be found in Appendix B.

Implementation Settings For all LLM-related models in our experiments, we employ the Qwen2.5-7B-Instruct (Team, 2024) as the LLM backbone and apply LoRA (Hu et al., 2022) to the attention layers of parameter-efficient fine-tuning. For the GNN component, we set the neighborhood aggregation depth to $K = 2$. We randomly split all three datasets into training, validation, and testing sets with a ratio of 40%, 20%, and 40%. The end-to-end optimization is realized using the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 3×10^{-4} across all three datasets. We conduct our experiments on a Linux system with 4 NVIDIA A800 GPUs (80GB memory each).

Evaluation Metrics We evaluate the experimental results over the three datasets under the ROC curve (AUC), Recall and Geometric Mean (G-Mean). LLMs generate outputs autoregressively based on token logits and sampling strategies. To compute the AUC, we define an anomaly score $\mathcal{S}_i$ for each node v_i. This score is derived from the log-likelihood margin between the fraudulent and normal target sequences:

$$\mathcal{S}_i = \log P(\mathbf{Y}^{(1)} \mid \mathbf{E}_{\text{input}}) - \log P(\mathbf{Y}^{(0)} \mid \mathbf{E}_{\text{input}}) \tag{16}$$

This formulation naturally accommodates multi-token sequences caused by diverse tokenization patterns and prompt designs. We evaluate Recall and G-Mean based on the final generated label $\hat{y}_i$.

4.2 Overall Evaluation

Table 3 summarizes the overall performance of all compared methods. Compared to classical general-purpose graph models including GCN, GAT, and GraphSAGE, the models specifically designed for graph fraud detection such as CARE-GNN, PC-GNN, BWGNN, and GTAN exhibit consistently superior performance across all three datasets. Since fraudulent targets often show structural camouflage and heterogeneous relational patterns, fraud-specific methods overcome these challenges from perspectives such as noise filtering and class distribution balancing. Our experimental results confirm their effectiveness and improvements. The Text-flattened LLMs exhibit severely degraded performance, unexpectedly lagging behind even the classical GNN baselines. This failure explicitly exposes the inherent semantic dilemma LLMs face

Method	Amazon			YelpChi			S-FFSD		
	AUC	Recall	G-Mean	AUC	Recall	G-Mean	AUC	Recall	G-Mean
GCN	0.8362	0.7751	0.7723	0.6338	0.5906	0.5931	0.7404	0.5777	0.6824
GAT	0.7954	0.7234	0.7296	0.5704	0.5363	0.5429	0.7051	0.5611	0.6583
GraphSAGE	0.8735	0.7842	0.8211	0.7374	0.6325	0.6146	0.6713	0.3814	0.6055
CARE-GNN	0.9200	0.8635	0.8824	0.7727	0.7046	0.7044	0.7112	0.4870	0.6230
PC-GNN	0.9665	0.9018	0.9062	0.8184	0.7348	0.7290	0.7033	0.5705	0.6213
BWGNN	0.9717	0.7872	0.8840	0.8876	0.6579	0.7695	0.7029	0.6201	0.6199
GTAN	0.9585	0.7561	0.8680	0.9172	0.5600	0.7363	0.8299	0.6921	0.6196
RGTAN	0.9638	0.8898	0.8836	0.9194	0.7743	0.7506	0.8246	0.6940	0.6279
Zero-shot LLM	0.5062	0.0321	0.1453	0.4145	0.0172	0.0284	0.4633	0.0304	0.1729
Instruct-LLM	0.8390	0.9077	0.7665	0.7124	0.6710	0.7444	0.7188	0.4408	0.6412
w/o LLM	0.9087	0.7812	0.8812	0.7726	0.5947	0.7395	0.7638	0.4912	0.6877
w/o Semantics	0.9640	0.8863	0.9197	0.9095	0.7775	0.6418	0.8364	0.7432	0.7801
w/o Joint-Opt	0.9703	**0.9271**	0.9233	0.9076	0.7915	0.8296	0.8410	0.7476	0.7588
LGSPF	**0.9805**	0.9179	**0.9310**	**0.9208**	**0.8346**	**0.8325**	**0.8719**	**0.7561**	**0.7837**

Table 3: Overall performance comparison and ablation studies across three benchmark datasets. All reported results are the averages of 5 independent runs under the same random seeds.

in Weak-TAG graphs: directly serializing high-dimensional floating-point arrays into discrete text prompts forces standard tokenizers to fragment continuous structural features.

Our proposed LGSPF achieves superior performance, consistently outperforming all baselines by a significant margin across the three datasets. LGSPF elegantly resolves the aforementioned semantic dilemma. By mapping GNN-extracted topologies into continuous soft prompts and bridging them with hybrid natural language templates, our framework successfully aligns structural features with the LLM's cognitive space without information degradation. While traditional task-specific GNNs struggle to explicitly interpret the high-order semantic differences among complex multi-relational topologies, LGSPF overcomes this bottleneck by leveraging the LLM's deep semantic comprehension, ultimately leading to a comprehensive and robust enhancement in graph fraud detection tasks.

4.3 Ablation Studies

In Table 3, we also conduct three ablation experiments to evaluate the effectiveness of the key components and designs in LGSPF. To validate the impact of the semantic reasoning and soft prompt, we introduce two ablation variants. In w/o LLM, we substitute the LLM backbone with a simple MLP classifier, where the outputs from the parallel GNN encoders are directly concatenated. In w/o Semantics, we remove all natural language instructions and relational descriptions included in Eq. 8, leaving only the isolated continuous graph tokens $s_1, \ldots, s_m$. Both variants result in a clear performance drop, which proves that the LLM possesses superior discriminative capacity for multi-relational representations. Furthermore, the textual descriptions serve as vital semantic anchors, effectively unlocking the LLM's potential to utilize its pre-trained real-world knowledge for enhanced fraud identification. To validate the necessity of end-to-end optimization, we design w/o Joint-Opt by cutting off the gradient backpropagation chain at the LLM-GNN interface. Specifically, we solely fine-tune the LLM while keeping the parallel GNNs frozen with random initialization. Although the performance slightly declines, it achieves several sub-optimal results across all three datasets. This phenomenon highlights the strong fine-grained semantic comprehension of LLMs, which can detect fraud patterns even with non-calibrated structural embeddings. Nevertheless, the end-to-end optimization promotes a deeper cross-modal alignment, elevating the overall performance.

4.4 Sensitivity Studies

In this section, we study the parameter sensitivity of our proposed LGSPF by varying the scale of the LLM backbone and the neighborhood aggregation depth K for the parallel GNN encoders. The experimental results across the three detection datasets

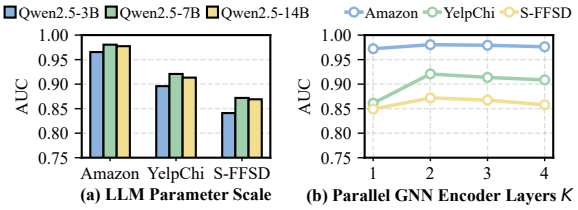

Figure 3: Sensitivity analysis of the LLM parameter scale and the aggregation depth K across three benchmark datasets evaluated by AUC.

Embedded View	AUC	Recall	G-Mean
Single-view R-S-R	0.8680	0.7043	0.7620
Single-view R-T-R	0.8531	0.8125	0.7187
Single-view R-U-R	0.8927	0.7504	0.8049
Single-view Average*	0.8713	0.7557	0.7619
Full view (LGSPF)	**0.9208**	**0.8346**	**0.8325**

Table 4: Interpretability study on the YelpChi dataset. The Single-view Average* represents the arithmetic mean of the three individual relation metrics.

are reported in Figure 3. Specifically, we vary the parameter scale of the Qwen2.5 series backbone among {3B, 7B, 14B} and adjust the depth K from 1 to 4. Regarding the capacity of the LLM backbone, we observe a significant performance improvement when upgrading the scale from 3B to 7B, while further scaling the LLM to 14B yields saturation or slight degradation. This indicates that a threshold of parameter scale is essential to comprehend complex structural semantics; however, it also implies that massive models might suffer from slight overfitting during parameter-efficient fine-tuning. Ultimately, the 7B model strikes the optimal balance between reasoning capability and adaptation efficiency, making it the most cost-effective choice for our framework. Regarding the structural receptive field, performance peaks at $K = 2$ and declines for $K \geq 3$. While $K = 1$ restricts the receptive field and misses higher-order topological fraud patterns, excessive depths ($K \geq 3$) invariably encounter the over-smoothing issue. Therefore, a moderate depth ($K = 2$) is essential to balance sufficient topological aggregation with the sharp discriminability of fraudulent nodes.

4.5 Interpretability Studies

To explore the potential of LGSPF in advancing the interpretability of Graph Fraud Detection, we conduct a single-relation embedding experiment on the YelpChi dataset. Specifically, we degenerate the comprehensive multi-relational prompt sequence by replacing the entire relation block $\{s_k, \mathbf{T}_{\text{desc}_k}\}_{k=1}^{m}$ with a single embedding-description pair $\{s_j, \mathbf{T}_{\text{desc}_j}\}$. As reported in Table 4, the detection performance fluctuates across different isolated relations. For instance, the R-U-R view achieves the highest AUC among three single views and the R-T-R view exhibits an advantage in Recall. This result demonstrates that distinct relational connections imply differentiated structural semantics. However, the integrated Full view decisively outperforms the Single-view Average* by 5.4%, 10.4%, and 8.5% in terms of AUC, Recall, and G-Mean, respectively. By prompting the LLM to process diverse relations simultaneously, LGSPF effectively mines the intrinsic correlations among multiple relations, achieving a deeper cross-semantic fusion that boosts detection performance. Furthermore, the high modularity of LGSPF allows for dynamic adjustments regarding the type and number of injected views based on domain-specific knowledge. This flexibility provides a powerful tool for diagnosing fraudulent mechanisms in different application contexts, shedding new light on advancing graph fraud detection from black-box predictions to behavior recognition.

5 Conclusion

In this paper, we propose LGSPF, a novel LLM-GNN Soft Prompt Framework for fraud detection. To overcome the challenge of deploying LLMs in Weak Text-Attributed Graph (Weak-TAG) scenarios where rich textual data is unavailable, we implement a continuous soft prompting mechanism that bridges the graph topology with the LLM's semantic space, successfully bypassing the feature distortion caused by discrete textualization. Furthermore, to address the multi-relational complexity inherent in fraudulent activities and the structural-semantic misalignment, we design a parallel GNN encoder alongside an end-to-end optimization paradigm. This architecture empowers the LLM to dynamically reason over distinct relational topologies. Extensive experiments across multiple benchmarks demonstrate that LGSPF achieves state-of-the-art performance. Beyond empirical superiority, our framework significantly enhances the semantic interpretability of fraud behaviors, providing a highly promising paradigm for advancing fraud detection from black-box predictions to transparent behavior recognition in privacy-sensitive domains.

Limitations

Although our proposed LGSPF framework demonstrates remarkable performance in multi-relational Graph Fraud Detection under the Weak-TAG setting, several limitations remain to be addressed in future research. First, regarding scalability, while we employ Parameter-Efficient Fine-Tuning and lightweight parallel GNN encoders to minimize resource consumption, the integration of Large Language Models inevitably introduces substantial computational overhead. Consequently, directly deploying LGSPF on ultra-large-scale industrial graph datasets with billions of nodes and edges remains computationally demanding, particularly in resource-constrained environments. Secondly, the framework's performance intrinsically depends on the cognitive reasoning capabilities of the pre-trained LLM backbone. Due to hardware limitations, our main experiments are conducted using Qwen2.5-7B model. This constraint may prevent the framework from fully unlocking and deeply mining more complex structural semantics. In the future, we aim to validate our framework on larger-scale foundational models to explore performance across more diverse fraud detection scenarios. Finally, the current implementation of LGSPF is primarily tailored for static multi-relational graphs. In real-world applications, fraudulent nodes and interactive behaviors frequently evolve over time. The absence of temporal dynamics restricts the model from capturing the chronological progression of camouflage strategies. In future work, we plan to extend our framework to dynamic graphs by incorporating temporal embeddings, thereby further improving its effectiveness and practical applicability.

References

He Cao, Zijing Liu, Xingyu Lu, Yuan Yao, and Yu Li. 2025. Instructmol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 354–379.

Runjin Chen, Tong Zhao, Ajay Jaiswal, Neil Shah, and Zhangyang Wang. 2024. Llaga: large language and graph assistant. In *Proceedings of the 41st International Conference on Machine Learning*, pages 7809–7823.

Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S Yu. 2020a. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 315–324.

Yingtong Dou, Guixiang Ma, Philip S Yu, and Sihong Xie. 2020b. Robust spammer detection by nash reinforcement learning. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 924–933.

Venus Haghighi, Behnaz Soltani, Nasrin Shabani, Jia Wu, Yang Zhang, Lina Yao, Quan Z Sheng, and Jian Yang. 2024. Tropical: Transformer-based hypergraph learning for camouflaged fraudster detection. In *2024 IEEE International Conference on Data Mining (ICDM)*, pages 121–130. IEEE.

Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30.

Li Han, Longxun Wang, Ziyang Cheng, Bo Wang, Guang Yang, Dawei Cheng, and Xuemin Lin. 2025. Mitigating the tail effect in fraud detection by community enhanced multi-relation graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*, 37(4):2029–2041.

Bryan Hooi, Hyun Ah Song, Alex Beutel, Neil Shah, Kijung Shin, and Christos Faloutsos. 2016. Fraudar: Bounding graph fraud in the face of camouflage. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 895–904.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3.

Chao Huang, Xubin Ren, Jiabin Tang, Dawei Yin, and Nitesh Chawla. 2024. Large language models for graphs: Progresses and directions. In *Companion Proceedings of the ACM Web Conference 2024*, pages 1284–1287.

Tairan Huang, Yili Wang, Qiutong Li, Changlong He, and Jianliang Gao. 2025. Can llms find fraudsters? multi-level llm enhanced graph fraud detection. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 1530–1538.

Xuanwen Huang, Yang Yang, Yang Wang, Chunping Wang, Zhisheng Zhang, Jiarong Xu, Lei Chen, and Michalis Vazirgiannis. 2022. Dgraph: A large-scale financial dataset for graph anomaly detection. *Advances in Neural Information Processing Systems*, 35:22765–22777.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 3045–3059.

Mengran Li, Pengyu Zhang, Wenbin Xing, Yijia Zheng, Klim Zaporojets, Junzhou Chen, Ronghui Zhang, Yong Zhang, Siyuan Gong, Jia Hu, Xiaolei Ma, Zhiyuan Liu, Paul Groth, and Marcel Worring. 2026a. A survey of large language models for data challenges in graphs. *Expert Systems with Applications*, 298:129643.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597.

Yuan Li, Jun Hu, Bryan Hooi, Bingsheng He, and Cheng Chen. 2026b. Dgp: A dual-granularity prompting framework for fraud detection with graph-enhanced llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 15171–15179.

Yuhan Li, Zhixun Li, Peisong Wang, Jia Li, Xiangguo Sun, Hong Cheng, and Jeffrey Xu Yu. 2024. A survey of graph meets large language model: progress and future directions. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, pages 8123–8131.

Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. 2021. Pick and choose: a gnn-based imbalanced learning approach for fraud detection. In *Proceedings of the web conference 2021*, pages 3168–3177.

Ziqi Liu, Chaochao Chen, Xinxing Yang, Jun Zhou, Xiaolong Li, and Le Song. 2018. Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 2077–2085.

Wai Weng Lo, Siamak Layeghy, Mohanad Sarhan, Marcus Gallagher, and Marius Portmann. 2022. E-graphsage: A graph neural network based intrusion detection system for iot. In *NOMS 2022-2022 IEEE/iFIP network operations and management symposium*, pages 1–9. IEEE.

Xiaoxiao Ma, Jia Wu, Shan Xue, Jian Yang, Chuan Zhou, Quan Z Sheng, Hui Xiong, and Leman Akoglu. 2021. A comprehensive survey on graph anomaly detection with deep learning. *IEEE transactions on knowledge and data engineering*, 35(12):12012–12038.

Julian John McAuley and Jure Leskovec. 2013. From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews. In *Proceedings of the 22nd international conference on World Wide Web*, pages 897–908.

Shebuti Rayana and Leman Akoglu. 2015. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th acm sigkdd international conference on knowledge discovery and data mining*, pages 985–994.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024a. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 491–500.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024b. Higpt: Heterogeneous graph language model. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 2842–2853.

Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. 2022. Rethinking graph neural networks for anomaly detection. In *International conference on machine learning*, pages 21076–21089. PMLR.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Haoyu Peter Wang, Shikun Liu, Rongzhe Wei, and Pan Li. 2025a. Generalization principles for inference over text-attributed graphs with large language models. In *Forty-second International Conference on Machine Learning*.

Jianyu Wang, Rui Wen, Chunming Wu, Yu Huang, and Jian Xiong. 2019. Fdgars: Fraudster detection via graph convolutional networks in online app review system. In *Companion proceedings of the 2019 World Wide Web conference*, pages 310–316.

Xiaodi Wang, Zhonglin Liu, Jiamiao Liu, and Jiayong Liu. 2023. Fraud detection on multi-relation graphs via imbalanced and interactive learning. *Information Sciences*, 642:119153.

Zehong Wang, Sidney Liu, Zheyuan Zhang, Tianyi Ma, Chuxu Zhang, and Yanfang Ye. 2025b. Can LLMs convert graphs to text-attributed graphs? In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1412–1432, Albuquerque, New Mexico. Association for Computational Linguistics.

Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Llmrec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM international conference on web search and data mining*, pages 806–815.

Yicong Wu, Guangyue Lu, Yuan Zuo, Huarong Zhang, and Junjie Wu. 2025. Graph-r1: Incentivizing the zero-shot graph learning capability in llms via explicit reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23920–23938.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24.

Sheng Xiang, Guibin Zhang, Dawei Cheng, and Ying Zhang. 2025. Enhancing attribute-driven fraud detection with risk-aware graph representation. *IEEE Transactions on Knowledge and Data Engineering*.

Sheng Xiang, Mingzhi Zhu, Dawei Cheng, Enxia Li, Ruihui Zhao, Yi Ouyang, Ling Chen, and Yefeng Zheng. 2023. Semi-supervised credit card fraud detection via attribute-driven graph representation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 14557–14565.

Han Xie, Da Zheng, Jun Ma, Houyu Zhang, Vassilis N Ioannidis, Xiang Song, Qing Ping, Sheng Wang, Carl Yang, Yi Xu, and 1 others. 2023. Graph-aware language model pre-training on a large graph corpus can help multiple graph applications. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5270–5281.

Haotian Xu, Yuning You, and Tengfei Ma. 2026. When structure doesn't help: Llms do not read text-attributed graphs as effectively as we expected. In *The Fourth Learning on Graphs Conference*.

Chengdong Yang, Hongrui Liu, Daixin Wang, Zhiqiang Zhang, Cheng Yang, and Chuan Shi. 2025. Flag: Fraud detection with llm-enhanced graph neural network. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5150–5160.

Ruosong Ye, Caiqi Zhang, Runhui Wang, Shuyuan Xu, and Yongfeng Zhang. 2024. Language is all a graph needs. In *Findings of the association for computational linguistics: EACL 2024*, pages 1955–1973.

Mengmei Zhang, Mingwei Sun, Peng Wang, Shen Fan, Yanhu Mo, Xiaoxiao Xu, Hong Liu, Cheng Yang, and Chuan Shi. 2024. Graphtranslator: Aligning graph model to large language model for open-ended tasks. In *Proceedings of the ACM Web Conference 2024*, pages 1003–1014.

Zhongjian Zhang, Xiao Wang, Mengmei Zhang, Jiarui Tan, and Chuan Shi. 2026. Toward graph-tokenizing large language models with reconstructive graph instruction tuning. In *Proceedings of the ACM Web Conference 2026*, pages 430–441.

A Datasets and Multi-Relational Construction

To guarantee the reproducibility of our experimental outcomes, we provide a thorough description regarding the background, raw node feature configuration, and multi-relational graph instantiation for the three benchmark datasets used in this study. Following the preprocessing paradigms outlined by Dou et al. (2020a) and Xiang et al. (2023), all unstructured attributes are discarded to enforce the Weak-TAG setting. The hand-crafted input feature dimensions for the Amazon, YelpChi, and S-FFSD datasets are rigorously specified as 25, 32, and 126, respectively.

A.1 Amazon Dataset

The Amazon dataset (McAuley and Leskovec, 2013) operates within the e-commerce domain, capturing user review dynamics centered around product entities under the *Musical Instruments* category. Following established conventions, users maintaining greater than 80% helpful upvotes are designated as benign entities ($y = 0$), while individuals with fewer than 20% helpful votes are categorized as fraudulent accounts ($y = 1$). We construct a multi-relational graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}_A)$ governed by three specific relationship types: $\mathcal{R}_A = \{r_{A1}, r_{A2}, r_{A3}\}$.

- **U-P-U Relation (r_{A1}):** Connects two reviewer nodes if they have authored a review for the identical product entity. This relation serves to isolate co-review target syndication.
- **U-S-U Relation (r_{A2}):** Connects two reviewer nodes if they assigned an identical star rating to any product within a tightly bounded temporal window of one week. It captures synchronized rating distortion.
- **U-V-U Relation (r_{A3}):** Connects two reviewer nodes if their textual review similarities reside within the top 5% of the global distribution, where similarity is computed via cosine distance over hand-crafted TF-IDF vectors.

A.2 YelpChi Dataset

The YelpChi dataset (Rayana and Akoglu, 2015) focuses on crowdsourced business intelligence, en-

compassing hotel and restaurant reviews processed by Yelp's proprietary anti-fraud filter. Review entities recommended as legitimate are denoted as normal ($y = 0$), whereas filtered reviews are tagged as spam/fraudulent ($y = 1$). Nodes are represented by 32-dimensional hand-crafted statistical features. We construct a multi-relational graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}_Y)$ governed by three specific relationship types: $\mathcal{R}_Y = \{r_{Y1}, r_{Y2}, r_{Y3}\}$.

- **R-T-R Relation (r_{Y1}):** Connects two review nodes if they correspond to the exact same business establishment and were posted within the identical calendar month, uncovering burst-mode review injection.
- **R-U-R Relation (r_{Y2}):** Connects two review nodes if they were generated by the same unique user account, tracking persistent fraudulent identity footprints.
- **R-S-R Relation (r_{Y3}):** Connects two review nodes if they share the same target business and hold identical star ratings, effectively capturing coordinated sentiment manipulation clusters.

A.3 S-FFSD Dataset

The Simulated Financial Fraud Semi-Supervised Dataset (S-FFSD) is an open-source, simulated, and scaled-down variant of the original proprietary FFSD dataset introduced by Xiang et al. (2023). It simulates an anonymous credit card transactional ecosystem where nodes denote individual financial transactions, and the numerical features are structured into a 126-dimensional dense vector space. Ground-truth labels are authenticated via real-world consumer reporting: transactions verified as fraudulent are marked as positive ($y = 1$), normal transactions as negative ($y = 0$), and unverified records are masked as unlabelled ($y = 2$).

Distinct from standard undirected graphs, the topological connections in S-FFSD are strictly governed by temporal dynamics. We decode the raw attributes *Source*, *Target*, *Location*, and *Type* to partition transactions into discrete groups. Within each specific group, nodes are sorted chronologically by their execution timestamps. Directed edges are subsequently constructed pointing from earlier transactions to later ones. To capture localized temporal dependencies while avoiding the computational bottleneck caused by dense over-connections, each transaction is explicitly connected only to the subsequent $k = 3$ transactions within the same group. Finally, self-loops are universally injected into the adjacency matrices to retain original node features during the GNN message-passing phase.

Following this temporal-directed paradigm, we construct a multi-relational graph structure $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R}_F)$ governed by four specific relationship types: $\mathcal{R}_F = \{r_{F1}, r_{F2}, r_{F3}, r_{F4}\}$.

- **Source Relation (r_{F1}):** Connects two transaction nodes, directed from the earlier transaction to the later one, if they originate from the exact same sender entity, tracking high-frequency velocity attacks from single compromised accounts.
- **Target Relation (r_{F2}):** Connects two transaction nodes, directed from the earlier transaction to the later one, if they share the identical receiver entity, facilitating the detection of merchant-side fraud collection points.
- **Location Relation (r_{F3}):** Connects two transaction nodes, directed from the earlier transaction to the later one, if they are executed within the identical geographical or electronic location identifier, isolating location-based card-cloning anomalies.
- **Type Relation (r_{F4}):** Connects two transaction nodes, directed from the earlier transaction to the later one, if they possess matching commercial transaction type codes, contextualizing the behavioral domain of fraudulent capital outflows.

B Prompt Designs

In this section, we present the exact prompt templates utilized in our experiments. Figure 4 details the multi-relational hybrid prompt structures employed by our proposed LGSPF framework across the three benchmark datasets. Figure 5 shows the serialization templates used by the Text-flattened LLM baselines.

LGSPF Prompt Template for Amazon Dataset

You are a fraud detection expert analyzing user review behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from user reviews of Amazon website products.
We apply the GraphSAGE algorithm to obtain node embeddings under different relational views as follows:
<|graph_pad_relation1|>: U-P-U relation embedding — connects users who reviewed the same product.
<|graph_pad_relation2|>: U-S-U relation embedding — connects users who gave the same star rating within a one-week window.
<|graph_pad_relation3|>: U-V-U relation embedding — connects users with top 5% mutual review text similarity based on TF-IDF.
Q: Is this review node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: [fraud / normal]

LGSPF Prompt Template for YelpChi Dataset

You are a fraud detection expert analyzing user review behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from user reviews of hotels and restaurants on the Yelp platform.
We apply the GraphSAGE algorithm to obtain node embeddings under different relational views as follows:
<|graph_pad_relation1|>: R-T-R relation embedding — connects reviews posted on the same business within the same month, revealing temporal co-review behaviors.
<|graph_pad_relation2|>: R-U-R relation embedding — connects reviews written by the same user, capturing user-level review consistency and behavioral tendencies.
<|graph_pad_relation3|>: R-S-R relation embedding — connects reviews with the same star rating for the same business, reflecting rating alignment or potential opinion manipulation.
Q: Is this review node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: [fraud / normal]

LGSPF Prompt Template for S-FFSD Dataset

You are a fraud detection expert analyzing transaction behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from financial transaction records in the FFSD (Financial Fraud Semi-supervised Dataset).
We apply the GraphSAGE algorithm to obtain node embeddings under different relational views as follows:
<|graph_pad_relation1|>: Source relation embedding — connects each transaction to the next 3 transactions with the same Source, ordered by time.
<|graph_pad_relation2|>: Target relation embedding — connects each transaction to the next 3 transactions with the same Target, ordered by time.
<|graph_pad_relation3|>: Location relation embedding — connects each transaction to the next 3 transactions with the same Location, ordered by time.
<|graph_pad_relation4|>: Type relation embedding — connects each transaction to the next 3 transactions with the same Type, ordered by time.
Q: Is this transaction node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: [fraud / normal]

Figure 4: The hybrid natural language prompt templates implemented in LGSPF. The explicitly highlighted special tokens (e.g., <|graph_pad_relationX|>) serve as the cross-modal interfaces, which are dynamically replaced by continuous relation-specific structural embeddings output by the parallel GNN encoders during the forward pass.

Text-flattened LLM Baseline Template for Amazon Dataset

You are a fraud detection expert analyzing user review behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from user reviews of Amazon website products.
We summarize the objective node features and its 1-hop neighbor features under different relational views as follows:
[[Flattened Target Node Features]]: object node features.
[[Flattened Neighbor Features]]: U-P-U relation features — connects users who reviewed the same product.
[[Flattened Neighbor Features]]: U-S-U relation features — connects users who gave the same star rating within a one-week window.
[[Flattened Neighbor Features]]: U-V-U relation features — connects users with top 5% mutual review text similarity based on TF-IDF.
Q: Is this review node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: `[fraud / normal]`

Text-flattened LLM Baseline Template for YelpChi Dataset

You are a fraud detection expert analyzing user review behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from user reviews of hotels and restaurants on the Yelp platform.
We summarize the objective node features and its 1-hop neighbor features under different relational views as follows:
[[Flattened Target Node Features]]: object node features.
[[Flattened Neighbor Features]]: R-T-R relation features — connects reviews posted on the same business within the same month, revealing temporal co-review behaviors.
[[Flattened Neighbor Features]]: R-U-R relation features — connects reviews written by the same user, capturing user-level review consistency and behavioral tendencies.
[[Flattened Neighbor Features]]: R-S-R relation features — connects reviews with the same star rating for the same business, reflecting rating alignment or potential opinion manipulation.
Q: Is this review node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: `[fraud / normal]`

Text-flattened LLM Baseline Template for S-FFSD Dataset

You are a fraud detection expert analyzing transaction behavior on a graph dataset.
This dataset is a multi-relational graph dataset constructed from financial transaction records in the FFSD (Financial Fraud Semi-supervised Dataset).
We summarize the objective node features and its 1-hop neighbor features under different relational views as follows:
[[Flattened Target Node Features]]: object node features.
[[Flattened Neighbor Features]]: Source relation features — connects each transaction to the next 3 transactions with the same Source, ordered by time.
[[Flattened Neighbor Features]]: Target relation features — connects each transaction to the next 3 transactions with the same Target, ordered by time.
[[Flattened Neighbor Features]]: Location relation features — connects each transaction to the next 3 transactions with the same Location, ordered by time.
[[Flattened Neighbor Features]]: Type relation features — connects each transaction to the next 3 transactions with the same Type, ordered by time.
Q: Is this transaction node fraudulent or normal?
You must analyze the embedding patterns and respond with exactly one of these two words: 'fraud' or 'normal'
A: `[fraud / normal]`

Figure 5: The serialization prompt templates utilized by the Text-flattened LLM baselines. Contrast to LGSPF's continuous soft prompts, these baselines substitute the explicitly highlighted placeholders (**[[Flattened X Features]]**) directly with raw, discrete textual sequences composed of long arrays of numerical features, which induces semantic degradation and tokenization fragmentation during LLM reasoning.