CSRI-FGL: CROSS-CLIENT SEMANTIC RELATION INFERENCE FOR FEDERATED GRAPH LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Federated graph learning (FGL) enables collaborative learning over distributed graphs while keeping raw graph data local. However, fragmented and non-IID local subgraphs make it difficult for existing parameter-aggregation-based methods to capture transferable semantic connection patterns for recovering semantically meaningful missing links. To address this issue, we propose CSRI-FGL, a framework that learns semantic connection patterns for missing-link recovery via LLM-guided relation inference. It distills node-level textual semantics and pairwise textual consistency signals into federatively aggregated edge generators, enabling each client to infer informative links without sharing raw graph data. Specifically, CSRI-FGL first extracts semantic representations and textual consistency signals from local node texts using a client-side LLM module to supervise the construction of local-pattern-enhanced graphs. The server then aggregates client-side edge-generator parameters into a global generator that captures transferable connection patterns across clients, and each client applies this global generator only within its own node set to build a global-pattern-enhanced local graph. Finally, a dual-branch GNN decouples personalized node representation learning from shared semantic connection-pattern learning by separately modeling local-pattern-enhanced and global-pattern-enhanced graphs. Experiments on five node-classification benchmarks under non-IID client partitions demonstrate that CSRI-FGL consistently outperforms representative FGL baselines in overall node-classification performance while maintaining competitive personalized performance.

1 INTRODUCTION

Graph-structured data are ubiquitous in real-world applications, including citation networks, recommender systems, social platforms, and biological networks. Graph neural networks (GNNs) have shown strong capability in modeling relational dependencies and learning structure-aware representations (Kipf & Welling, 2017; Hamilton et al., 2017; Veličković et al., 2018). In many practical scenarios, however, graph data are distributed across multiple organizations, platforms, or devices and cannot be centralized due to privacy, security, and regulatory constraints. This raises a key question: how can multiple parties collaboratively train effective graph models while keeping raw graph structures, node attributes, and sensitive information local?

Federated graph learning (FGL) addresses this problem by combining federated optimization (McMahan et al., 2017; Kairouz et al., 2021) with the relational modeling power of GNNs. By exchanging model updates rather than raw graph data, FGL enables privacy-preserving collaborative graph learning. However, existing FGL methods mainly rely on parameter or representation aggregation and remain limited in transferring semantic connection knowledge across clients. Since each client observes only a fragmented and non-IID local subgraph, useful semantic connection patterns may be scattered across clients, making it difficult to recover semantically meaningful missing links for local message passing.

This limitation reveals a central challenge in federated graph learning: effective collaboration requires transferable connection knowledge beyond individual local graphs, whereas federated protocols prohibit direct exchange of raw graph structures, node attributes, labels, and textual information. A desirable solution should therefore transfer connection knowledge at the pattern level, rather than relying on explicit cross-client neighbor exchange or server-side global graph construction. In this

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

work, we focus on semantic connection patterns, namely reusable rules that characterize when two nodes are likely to be semantically connected according to their textual descriptions and local graph context. Importantly, we do not infer explicit cross-client edges. Instead, we transfer cross-client semantic relation knowledge through federatively learned edge generators, and each client applies the learned global connection pattern only within its own local node set.

To this end, we propose CSRI-FGL, a federated graph learning framework for cross-client semantic relation inference. Each client employs a frozen local LLM module as a semantic teacher to extract node-level semantic representations and pairwise textual consistency scores from local node texts. Rather than directly sharing LLM outputs or node-level semantic representations, CSRI-FGL distills these local semantic signals into a trainable semantic edge generator. The server then aggregates client-side edge-generator parameters into a global semantic edge generator without collecting raw texts, node embeddings, labels, or graph topology. After receiving the global generator, each client applies it only to candidate non-edges within its own node set to construct a global-pattern-enhanced local graph. A dual-branch GNN further separates client-specific local graph modeling from transferable semantic connection-pattern learning, enabling personalized prediction while benefiting from federated semantic collaboration.

The main contributions of this work are summarized as follows:

- We identify semantic connection pattern learning as an underexplored challenge in federated graph learning, where fragmented and non-IID local subgraphs hinder semantically meaningful missing-link recovery.
- We propose CSRI-FGL, an LLM-assisted framework that distills node-level textual semantics and pairwise consistency signals into federatively aggregated semantic edge generators, enabling transferable relation inference without raw graph-data exchange.
- We evaluate CSRI-FGL on five text-attributed node-classification benchmarks under label-skewed federated partitions. Results show consistent gains in both local personalization and collaborative generalization, while ablations verify the effects of semantic distillation, graph enhancement, and dual-branch decoupling.

2 Related Works

2.1 Federated Graph Learning

Federated graph learning (FGL) extends federated learning to graph-structured data, where each client maintains a private local graph and collaboratively trains graph models without exchanging raw node attributes or local topology (Fu et al., 2022; Liu et al., 2025). Compared with conventional federated learning, FGL faces additional challenges caused by structural heterogeneity, non-IID label and feature distributions, and partial observability of local neighborhoods.

Existing FGL methods address these challenges through federated optimization, structural knowledge sharing, personalization, and decentralized collaboration. Representative methods include FedAvg-style federated GNN training (McMahan et al., 2017), heterogeneity-aware federated optimization (Li et al., 2020), structural knowledge sharing (Tan et al., 2023), cluster-driven FGL (Caldarola et al., 2021), FedGL (Chen et al., 2024a), GraphFL (Wang et al., 2022), SubgraphFL (Zhang et al., 2021), FedEgo (Zhang et al., 2024), D-FedGNN (Pei et al., 2021), DFGL2GS (Huang et al., 2025), and recent personalized or invariant FGL methods (Tan et al., 2024; Wang et al., 2025). These methods improve collaborative training or personalization under graph heterogeneity, but mainly rely on parameter-level, representation-level, or structure-heuristic mechanisms. They do not explicitly distill semantic connection knowledge from node texts into transferable edge-generation patterns. CSRI-FGL addresses this gap by using client-side LLM signals to supervise semantic edge generators and by aggregating these generators into a global semantic connection pattern.

2.2 LLM for Graphs

Large language models (LLMs) and language-model-based methods have recently been used to enhance graph learning, especially for text-attributed graphs (Zhao et al., 2023; Chen et al., 2024b; Tang et al., 2024). Existing studies use language models or LLMs as semantic encoders or teachers

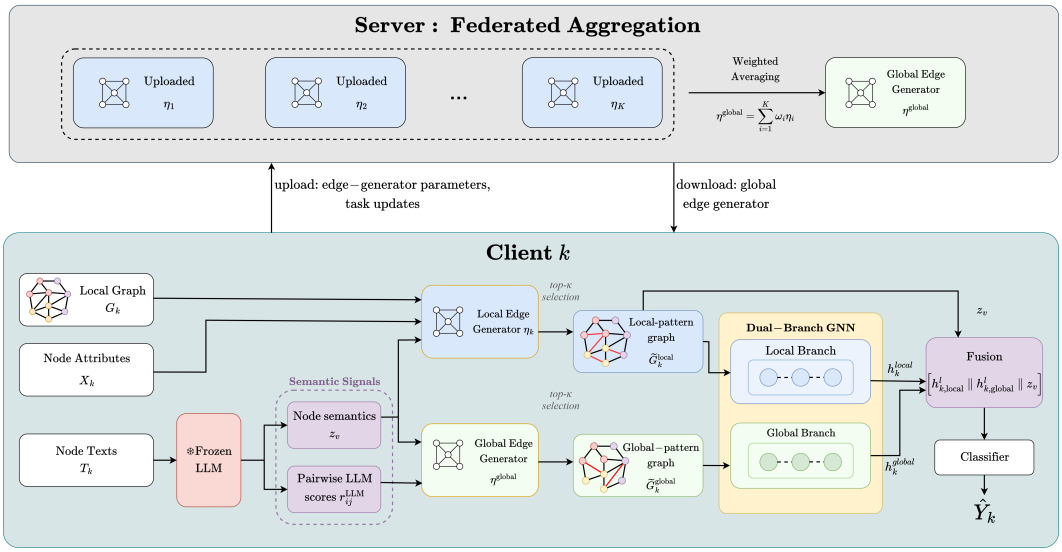

Figure 1: Overall framework of CSRI-FGL. The server aggregates uploaded edge-generator parameters into a global generator, while each client uses local graph data and LLM-derived semantic signals to construct local-pattern and global-pattern enhanced graphs, with top-κ_b selection applied to candidate non-edges in each branch. A dual-branch GNN then fuses the learned representations for node classification.

for text-attributed graph representation learning (Zhao et al., 2023; Chen et al., 2024b; He et al., 2024; Pan et al., 2024), develop instruction-tuned graph-LLM frameworks such as GraphGPT (Tang et al., 2024), study graph-to-text encoding for LLM reasoning (Fatemi et al., 2024), or exploit LLMs for graph structure learning and augmentation, such as GraphEdit and LLMRec (Guo et al., 2024; Wei et al., 2024). These methods improve semantic understanding and structure refinement in graph learning, but they are primarily designed for centralized settings, where graph texts, node attributes, or graph structures can be accessed during training or inference.

Under federated constraints, directly using LLMs to construct global graph structures or infer cross-client node relations is impractical, since raw texts, node attributes, labels, and local topology must remain on separate clients. CSRI-FGL therefore uses LLMs only as client-side semantic signal providers. The LLM-derived node representations and pairwise textual consistency scores are distilled into local semantic edge generators, while cross-client collaboration is achieved by aggregating edge-generator parameters rather than sharing raw texts, node-level representations, or cross-client neighbors.

3 Methodology

Problem Setting and Notation. This section first introduces the problem setting and basic notation. Consider a federated graph learning system with K clients, where each client holds a private graph dataset. The local graph of client k is denoted as

$$G_k = (\mathcal{V}_k, \mathcal{E}_k, X_k, T_k, Y_k), \tag{1}$$

where $\mathcal{V}_k$ denotes the local node set, $\mathcal{E}_k$ denotes the local edge set, X_k denotes node attributes, T_k denotes node textual descriptions, and Y_k denotes node labels. For node $v \in \mathcal{V}_k$, $x_v \in X_k$ denotes its attribute vector, $t_v \in T_k$ denotes its textual description, and $y_v \in Y_k$ denotes its label.

3.1 Framework Overview

As shown in Figure 1, CSRI-FGL follows a client–server federated learning protocol. In each communication round, every client first uses a frozen local LLM module to extract node-level semantic representations and pairwise textual consistency scores from its local node texts. These

signals supervise a client-side semantic edge generator $\eta_k(\cdot)$, which selects informative candidate non-edges within $\mathcal{V}_k$ and constructs a local-pattern-enhanced graph $\widetilde{G}_k^{\text{local}}$.

After local optimization, clients upload task-model updates and edge-generator parameters, while raw texts, node attributes, labels, local edge sets, prompts, LLM outputs, and enhanced edges remain local. The server aggregates the uploaded edge-generator parameters into a global generator $\eta^{\text{global}}(\cdot)$, which captures transferable semantic connection patterns without constructing a global graph or accessing cross-client neighbors. Each client applies this global generator only within its own node set to construct a global-pattern-enhanced graph $\widetilde{G}_k^{\text{global}}$. A dual-branch GNN then processes the local- and global-pattern-enhanced graphs separately and fuses their representations with LLM-derived semantic features for node classification.

3.2 Client-side Processing

3.2.1 LLM-based Semantic Enhancement

CSRI-FGL uses a frozen client-side LLM module to extract semantic signals for edge-generator learning. For each node $v \in \mathcal{V}_k$, client k constructs a textual description s_v from its local textual attributes and encodes it into a compact semantic representation:

$$h_v^{\text{text}} = \text{LLM}_{\text{enc}}^{\text{frz}}(s_v), \qquad z_v = \text{Norm}\big(P(h_v^{\text{text}})\big)\,, \tag{2}$$

where $P(\cdot)$ is a shared projection module and $\text{Norm}(\cdot)$ denotes normalization. The resulting representation z_v is used as the input to semantic edge generators. Since the LLM encoder is frozen and consistently deployed across clients, the semantic representation space remains aligned during federated training.

To provide pairwise semantic supervision, client k further estimates textual consistency for sampled local node pairs:

$$r_{ij}^{\text{LLM}} = \text{LLMScore}_k(s_i, s_j), \qquad i, j \in \mathcal{V}_k, \tag{3}$$

where $r_{ij}^{\text{LLM}} \in [0, 1]$ denotes the semantic consistency score used as soft supervision for training the local semantic edge generator $\eta_k(\cdot)$. All LLM-related operations are performed locally, and only task-model updates and edge-generator parameters are communicated during federated training. Supplementary Material, Sections A.1–A.3 provides score-calibration, relation-quality, and node-versus-pair diagnostics.

3.2.2 Semantic Connection Pattern Learning and Graph Enhancement

Given the compact semantic representation z_v and the pairwise semantic supervision r_{ij}^{LLM}, client k learns a semantic edge generator $\eta_k(\cdot)$ to model local semantic connection patterns. For a local node pair (i, j), the local semantic connection score is

$$\widehat{y}_{ij}^{\text{local}} = \eta_k(\phi(z_i, z_j))\,, \qquad i, j \in \mathcal{V}_k, \tag{4}$$

where $\phi(z_i, z_j)$ denotes a pairwise semantic representation. For undirected graphs, $\phi(\cdot, \cdot)$ is symmetric. The edge generator is trained using observed local edges, sampled non-edges, and LLM-derived semantic supervision, with the objective defined in Section 3.5.

For node i, client k constructs a candidate non-edge set

$$\mathcal{C}_k(i) \subseteq \{j \mid j \in \mathcal{V}_k,\ j \neq i,\ (i, j) \notin \mathcal{E}_k\}. \tag{5}$$

Each candidate pair is scored by combining structural proximity and semantic connectivity. To avoid redundant definitions, we use $b \in \{\text{local}, \text{global}\}$ to denote the local and global enhancement branches:

$$s_{ij}^b = \alpha \frac{|\mathcal{N}_k(i) \cap \mathcal{N}_k(j)|}{\sqrt{|\mathcal{N}_k(i)|\,|\mathcal{N}_k(j)|} + \epsilon} + (1 - \alpha)\widehat{y}_{ij}^b, \tag{6}$$

where $\widehat{y}_{ij}^{\text{local}}$ is produced by $\eta_k(\cdot)$ and $\widehat{y}_{ij}^{\text{global}}$ is produced by the global edge generator $\eta^{\text{global}}(\cdot)$ received from the server. The coefficient α controls the balance between local structural proximity and semantic connection scores.

The enhanced semantic edge set is selected by a per-node top-κ_b rule:

$$\mathcal{E}_k^b = \left\{(i,j) \;\middle|\; j \in \mathrm{Top}_{\kappa_b}\big(\mathcal{C}_k(i); s_{iu}^b\big),\ s_{ij}^b \geq \tau_b\right\}, \quad b \in \{\mathrm{local}, \mathrm{global}\}. \tag{7}$$

The corresponding enhanced local graph is

$$\widetilde{G}_k^b = \left(\mathcal{V}_k, \mathcal{E}_k \cup \mathcal{E}_k^b\right), \quad b \in \{\mathrm{local}, \mathrm{global}\}. \tag{8}$$

The local-pattern-enhanced graph $\widetilde{G}_k^{\mathrm{local}}$ is constructed using the client-side generator $\eta_k(\cdot)$, while the global-pattern-enhanced graph $\widetilde{G}_k^{\mathrm{global}}$ is constructed by applying $\eta^{\mathrm{global}}(\cdot)$ only to candidate non-edges within $\mathcal{V}_k$. For undirected graphs, selected semantic edges are symmetrized before graph construction. Both enhanced edge sets remain local and are never transmitted to the server. Since candidate pairs are constructed only within $\mathcal{V}_k$, the generated semantic edges never connect nodes from different clients. Therefore, $\widetilde{G}_k^{\mathrm{local}}$ and $\widetilde{G}_k^{\mathrm{global}}$ should be interpreted as client-local graph views guided by different semantic connection patterns, rather than as approximations of a server-side global graph.

3.3 Server-side Semantic Connection Pattern Aggregation

After local optimization, client k uploads the task-model updates and the parameters $\phi_{\eta,k}^{(t)}$ of its local semantic edge generator $\eta_k(\cdot)$ at communication round t. Since all clients share the same edge-generator architecture, the server aggregates client-side generators by weighted averaging:

$$\begin{aligned} \phi_\eta^{\mathrm{global},(t+1)} &= \sum_{k=1}^{K} \omega_k^{(t)} \phi_{\eta,k}^{(t)}, \\ \eta^{\mathrm{global},(t+1)}(\cdot) &= \eta\Big(\cdot; \phi_\eta^{\mathrm{global},(t+1)}\Big). \end{aligned} \tag{9}$$

Here $\omega_k^{(t)}$ denotes the aggregation weight of client k. In the main experiments, we use size-based aggregation with $\omega_k^{(t)} = N_k/N$, where $N_k = |\mathcal{V}_k^{\mathrm{tr}}|$ and $N = \sum_{k=1}^{K} N_k$. The server broadcasts the aggregated task-model parameters and $\phi_\eta^{\mathrm{global},(t+1)}$ back to clients. Each client then instantiates $\eta^{\mathrm{global},(t+1)}(\cdot)$ and applies it only to candidate non-edges within its own node set $\mathcal{V}_k$, following the graph-enhancement rule in Equation (7). The resulting $\widetilde{G}_k^{\mathrm{global}}$ is constructed locally and used by the global branch to model transferable semantic connection patterns. This design keeps the communicated information at the model-parameter level. The global generator captures reusable semantic connection rules rather than concrete client-specific graph structures. As a result, CSRI-FGL transfers relation patterns across clients while keeping node-level graph enhancement entirely local.

3.4 Dual-branch Graph Neural Network

To handle client heterogeneity, CSRI-FGL adopts a dual-branch GraphSAGE architecture (Hamilton et al., 2017). The local branch operates on the local-pattern-enhanced graph $\widetilde{G}_k^{\mathrm{local}}$, while the global branch operates on the global-pattern-enhanced graph $\widetilde{G}_k^{\mathrm{global}}$. These two graph views share the same local node set $\mathcal{V}_k$ but contain semantic edges generated from different connection patterns. This design separates client-specific connection modeling from globally transferable connection-pattern learning.

For each branch $b \in \{\mathrm{local}, \mathrm{global}\}$, the initial node representation combines the raw node attribute and the LLM-derived semantic representation:

$$x_v^0 = R([x_v \| z_v]), \qquad H_k^b = \mathrm{GNN}_b\Big(X_k^0, \widetilde{G}_k^b\Big), \tag{10}$$

where $R(\cdot)$ is an input projection module and $\mathrm{GNN}_b(\cdot)$ denotes the branch-specific GraphSAGE encoder.

After L layers, the node representation from the local branch, the global branch, and the compact semantic representation are fused for prediction:

$$H_v = [h_{\mathrm{local},v}^L \| h_{\mathrm{global},v}^L \| z_v], \qquad \widehat{y}_v = \mathrm{softmax}(f_{\mathrm{cls}}(H_v)). \tag{11}$$

Here, the local branch captures client-specific structural-semantic patterns, whereas the global branch captures transferable patterns induced by the federated semantic edge generator. The fused representation preserves both graph message-passing information and LLM-derived textual semantics.

Table 1: Dataset statistics.

Dataset	Nodes	Edges	Features	Classes
Citeseer	3,327	4,732	3,703	6
PubMed	19,717	44,338	500	3
Wiki	2,405	17,981	4,973	17
Amazon-Computers	13,752	491,722	767	10
Amazon-Photo	7,650	238,162	745	8

3.5 Multi-task Optimization

Client k jointly optimizes node classification, branch-level consistency, federated regularization, and semantic edge-generator learning. At communication round t, the local objective is defined as

$$\mathcal{L}_k^{(t)} = \mathcal{L}_{\mathrm{CE}}^{(k)} + \lambda_1 \mathcal{L}_{\mathrm{cons}}^{(k)} + \lambda_2 \mathcal{L}_{\mathrm{FedReg}}^{(k)} + \lambda_3 \mathcal{L}_{\mathrm{edge}}^{(k)}, \tag{12}$$

where $\mathcal{L}_{\mathrm{CE}}^{(k)}$ is the standard cross-entropy loss over labeled training nodes, and λ_1, λ_2, and λ_3 control the weights of the auxiliary objectives.

To stabilize transferable representation learning, we softly align the global branch with the personalized local branch and constrain transferable parameters from drifting far from the server-aggregated references:

$$\begin{aligned} \mathcal{L}_{\mathrm{cons}}^{(k)} &= \frac{1}{|\mathcal{V}_k^{\mathrm{tr}}|} \sum_{v \in \mathcal{V}_k^{\mathrm{tr}}} \left\| h_{\mathrm{global},v}^{L} - \mathrm{sg}\big(h_{\mathrm{local},v}^{L}\big) \right\|_2^2, \\ \mathcal{L}_{\mathrm{FedReg}}^{(k)} &= \left\| \Theta_{\mathrm{global},k}^{(t)} - \overline{\Theta}_{\mathrm{global}}^{(t)} \right\|_2^2 + \rho \left\| \phi_{\eta,k}^{(t)} - \phi_{\eta}^{\mathrm{global},(t)} \right\|_2^2. \end{aligned} \tag{13}$$

Here, $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator, $\overline{\Theta}_{\mathrm{global}}^{(t)}$ denotes the global-branch reference parameters, and ρ controls the regularization strength on the semantic edge generator. Supplementary diagnostic analyses of reliability-controlled edge variants and alternative fusion strategies, which are not part of the main configuration, are provided in Supplementary Material, Sections A.4 and A.5, respectively.

The edge-generator objective combines structural supervision from observed and sampled edges with LLM-derived semantic distillation:

$$\begin{aligned} \mathcal{L}_{\mathrm{edge}}^{(k)} &= \mathcal{L}_{\mathrm{BCE}}^{(k)} + \gamma \mathcal{L}_{\mathrm{distill}}^{(k)}, \\ \mathcal{L}_{\mathrm{distill}}^{(k)} &= \frac{1}{|\mathcal{P}_k|} \sum_{(i,j) \in \mathcal{P}_k} \left\| \eta_k(\phi(z_i, z_j)) - r_{ij}^{\mathrm{LLM}} \right\|_2^2. \end{aligned} \tag{14}$$

$\mathcal{L}_{\mathrm{BCE}}^{(k)}$ is the binary cross-entropy loss computed over observed local edges and sampled local non-edges, $\mathcal{P}_k$ denotes the sampled node-pair set with LLM-derived semantic consistency scores, and γ controls the contribution of semantic distillation.

4 Experiments

4.1 Experimental Setup

Datasets, Metrics, and Partitions. We evaluate CSRI-FGL on five node-classification benchmarks covering citation networks, Wikipedia graphs, and e-commerce co-purchase graphs: Citeseer, PubMed, Wiki, Amazon-Computers, and Amazon-Photo. Table 1 summarizes the dataset statistics.

Macro-F1 is used as the primary metric under non-IID label distributions. We evaluate each method under two protocols: **Local Test**, which measures personalization on each client's held-out local test split, and **Collaborative Test**, which measures cross-client generalization on a balanced test split reserved before federated partitioning. All methods follow the same transductive node-classification protocol with label masking, and test labels are never used during training.

We simulate non-IID clients using label-skew partitions with $K \in \{3, 6, 10\}$ for the main comparison and $K = 6$ for ablation and sensitivity analyses. Before client partitioning, 30% of nodes are reserved as a balanced collaborative test split, while the remaining nodes are assigned to clients with heterogeneous class coverage. Each client's local nodes are further split into train/validation/local-test sets with an 80%/10%/10% stratified ratio. Reserved test nodes are excluded from training-time message passing and graph enhancement to avoid evaluation leakage. Additional partitioning and implementation details are provided in Supplementary Material, Sections B.1 and B.2. Unless otherwise specified, CSRI-FGL uses Mistral-7B-Instruct as the frozen client-side semantic encoder, and all LLM-derived node representations and pairwise consistency scores are computed offline and cached locally before federated training.

Baseline Methods. We compare CSRI-FGL with representative federated graph learning baselines covering local training, standard federated aggregation, decentralized collaboration, privacy-aware personalization, and representation disentanglement.

Local-only trains an independent GraphSAGE model on each client without federated communication. It serves as a non-collaborative reference for measuring the benefit of cross-client learning. FedSage applies FedAvg (McMahan et al., 2017) to a GraphSAGE backbone (Hamilton et al., 2017), and is used as the standard federated GNN baseline. FedGCN applies FedAvg to a GCN backbone (Kipf & Welling, 2017), providing a comparison with another commonly used GNN encoder. GraphFL (Wang et al., 2022) is included as a representative federated graph learning method for semi-supervised node classification under non-IID graph data. D-FedGNN (Pei et al., 2021) represents decentralized federated GNN collaboration without relying on a standard centralized aggregation server. FedEgo (Zhang et al., 2024) is included as a privacy-aware personalized federated graph learning baseline based on ego-graph information. DFGL2GS (Huang et al., 2025) is used as a strong recent baseline that disentangles personalized and generalized graph representations in federated graph learning.

For fair comparison, all methods are evaluated under the same datasets, client partitions, train/validation/test splits, communication budget, optimizer, number of communication rounds, number of local epochs, and evaluation metrics. Unless a baseline requires a method-specific architecture, we use a 2-layer GraphSAGE encoder with hidden dimension 256. For method-specific baselines such as FedGCN, the original encoder type is retained while matching the hidden dimension and training budget as closely as possible. All baselines follow the same federated protocol: each client can access only its own local graph data, and no method is allowed to access raw node features, node texts, labels, local edge sets, or concrete graph structures from other clients.

Unless explicitly stated, baseline methods do not use LLM-derived semantic signals, semantic edge generators, or semantic graph enhancement modules. This setting isolates the contribution of the proposed client-side LLM-assisted semantic relation inference and federated edge-generator aggregation mechanism. To further examine the effect of different components, we provide controlled ablations in Section 4.3. In the ablation study, Baseline denotes FedAvg with a 2-layer GraphSAGE backbone, without LLM-derived semantic signals, semantic edge generation, or graph enhancement.

4.2 Main Results

4.2.1 Performance Comparison

Table 2 reports the main results on all datasets. We vary the number of clients ($K \in \{3, 6, 10\}$) and report the mean macro-F1 averaged over these client configurations and five random seeds.

Key Observations. CSRI-FGL achieves the best performance on all Collaborative Test datasets and maintains strong Local Test performance. On Local Test, it ranks first on Citeseer, Wiki, and Amazon-Photo, and remains competitive on Amazon-Computers and PubMed. These results indicate that LLM-guided semantic graph enhancement and dual-branch modeling can support personalized prediction under label-skewed client distributions. On Collaborative Test, CSRI-FGL outperforms DFGL2GS across all datasets, with absolute macro-F1 improvements of 0.011–0.020. These gains indicate that aggregating client-side semantic edge generators into a global edge generator transfers useful semantic connection patterns beyond standard parameter aggregation and representation disentanglement. The improvements are averaged over $K \in \{3, 6, 10\}$, suggesting that CSRI-FGL remains effective under different degrees of client fragmentation. The local-pattern branch preserves client-specific structural

Table 2: Macro-F1 scores for node classification under local personalization and Collaborative Test evaluation.

(a) Local Test (Personalization)

Method	Cite.	Wiki	A-Comp	Pub.	A-Photo
Local-only	0.759	0.893	0.909	0.897	0.736
GraphFL	0.837	0.838	0.922	0.874	0.799
FedSage	0.859	0.907	0.925	0.915	0.836
FedGCN	0.808	0.820	0.930	0.907	0.789
D-FedGNN	0.888	0.902	0.932	0.927	0.863
FedEgo	0.908	0.927	0.974	0.944	0.882
DFGL2GS	0.896	0.947	**0.978**	**0.959**	0.872
CSRI-FGL	**0.910**	**0.950**	0.965	0.955	**0.885**

(b) Cross-client Test

Method	Cite.	Wiki	A-Comp	Pub.	A-Photo
Local-only	0.670	0.734	0.780	0.792	0.649
GraphFL	0.701	0.683	0.842	0.831	0.687
FedSage	0.704	0.727	0.837	0.823	0.692
FedGCN	0.709	0.694	0.810	0.832	0.696
D-FedGNN	0.695	0.676	0.810	0.849	0.684
FedEgo	0.715	0.802	0.883	0.872	0.702
DFGL2GS	0.724	0.809	0.883	0.882	0.712
CSRI-FGL	**0.742**	**0.823**	**0.895**	**0.893**	**0.732**

Table 3: Comparison of frozen semantic encoders under the default setting $K = 6$ with a fixed prompt template (mean macro-F1 across five benchmarks). Mistral-7B denotes Mistral-7B-Instruct, and Qwen2.5-7B denotes Qwen2.5-7B-Instruct.

LLM Backbone	Local Test	Collaborative Test
LLaMA2-7B	0.9164	0.7764
Vicuna-7B	0.9382	0.7976
Qwen2.5-7B	0.9470	0.8150
Mistral-7B	**0.9496**	**0.8164**

and semantic preferences, while the global-pattern branch introduces transferable connection patterns learned through federated edge-generator aggregation.

4.2.2 Semantic Encoder Choice under a Fixed Prompt

Table 3 compares frozen 7B-scale semantic encoders under the default setting $K = 6$. All non-LLM components, the training budget, and the pairwise-scoring prompt are fixed, with no backbone-specific prompt tuning; the results are therefore not directly identical to Table 2.

Analysis. Mistral-7B-Instruct (Jiang et al., 2023) achieves the highest mean macro-F1 under both protocols (0.9496 for Local Test and 0.8164 for Collaborative Test) and is therefore used by default. The margin over Qwen2.5-7B is small, so this result reflects the current English text-attributed benchmarks and fixed prompt, rather than intrinsic or universal superiority; CSRI-FGL remains model-agnostic. The prompt and parsing rule for $\mathrm{LLMScore}$ are provided in Supplementary Material, Section B.3.

4.3 Ablation Study

Ablations isolate LLM semantic modeling, local and global graph enhancement, and dual-branch modeling under $K = 6$ on Citeseer, PubMed, and Wiki. Baseline is FedAvg with a 2-layer GraphSAGE backbone; successive variants add frozen LLM embeddings, local-pattern enhancement, and global-pattern enhancement in a single branch. Full CSRI-FGL uses both graph views with the dual-branch GNN.

4.3.1 Ablation Analysis

Table 4 shows that each component contributes to CSRI-FGL. Adding LLM semantic embeddings improves the FedAvg GraphSAGE baseline, indicating that textual semantics provide useful node-level signals under non-IID client partitions. Local-pattern enhancement further improves both Local Test and Collaborative Test performance, showing that client-side semantic edge generation benefits personalized and cross-client evaluation.

The variant using global-pattern enhancement without dual-branch decoupling performs worse on Collaborative Test, suggesting that directly mixing local and global-pattern-guided graph signals can introduce negative transfer under label skew. The full model achieves the best results on both

Table 4: Ablation study under the default setting $K = 6$ (macro-F1 averaged over Citeseer, PubMed, and Wiki).

Configuration	Local Test	Collaborative Test
Baseline	0.878	0.739
+ LLM only	0.900	0.759
+ Local-pattern	0.915	0.768
Global w/o Dual-branch	0.906	0.734
Full CSRI-FGL	**0.9383**	**0.8193**

protocols, confirming the benefit of separating personalized local graph modeling from transferable semantic connection-pattern learning. Full dataset-level ablation results are provided in Supplementary Material, Section B.4.

4.3.2 Sensitivity and Convergence Analysis

We further examine the sensitivity of CSRI-FGL to the top-κ_l edge selection parameter and the structure–semantic balance parameter α, as well as its convergence behavior under the same communication budget. As shown in Supplementary Material, Sections B.5 and B.6, CSRI-FGL remains stable under moderate hyperparameter perturbations and converges consistently on both Local Test and Collaborative Test.

5 Conclusion

CSRI-FGL distills client-side LLM signals into semantic edge generators, aggregates these generators, and uses locally constructed local- and global-pattern graph views in a dual-branch GNN. Across five benchmarks, it improves collaborative performance while retaining competitive personalized performance without exchanging raw texts, attributes, labels, or edge sets.

Limitations and future work. CSRI-FGL relies on informative node texts and frozen LLM-derived semantic signals, so its effectiveness may degrade when textual attributes are noisy, missing, or weakly aligned with graph structure. Although raw texts, node attributes, labels, local edges, and enhanced edges are not exchanged, the current framework does not provide formal privacy guarantees such as differential privacy or secure aggregation. Future work will improve robustness under limited textual attributes and extend semantic connection pattern learning to more diverse graph tasks.

AI use statement

Frozen LLMs are used as semantic modules in the proposed method, as described in Sections 3.2 and 4. Generative AI tools were also used to assist with translation, language editing, LaTeX formatting, and the presentation of Figure 1 (including explanatory top-κ selection annotations). All AI-assisted text and figures were reviewed by the authors, who take responsibility for the final manuscript, including its text, claims, and figures.

Reproducibility statement

To support reproducibility, Section 4.1 reports the datasets, evaluation protocols, client partitions, and common training budget. Section 4.2.2 specifies the frozen semantic encoders and fixed-prompt comparison, while Section 4.3 summarizes the controlled ablation setting. Supplementary Material, Sections B.1–B.6 provides the partitioning and implementation details, the $\mathrm{LLMScore}$ prompt and parsing rule, dataset-level ablations, and the sensitivity and convergence analyses. Main results are averaged over five random seeds and $K \in \{3, 6, 10\}$; controlled analyses use $K = 6$ unless stated otherwise. All LLM-derived semantic representations and pairwise scores are computed offline and cached locally before federated training, keeping the stochastic training and evaluation protocol separate from LLM querying.

REFERENCES

Debora Caldarola, Massimiliano Mancini, Fabio Galasso, Marco Ciccone, Emanuele Rodola, and Barbara Caputo. Cluster-driven graph federated learning over multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 2749–2758. IEEE, 2021.

Chuan Chen, Ziyue Xu, Weibo Hu, Zibin Zheng, and Jie Zhang. FedGL: Federated graph learning framework with global self-supervision. *Information Sciences*, 657:119976, 2024a.

Zhikai Chen, Haitao Mao, Hang Li, Wei Jin, Hongzhi Wen, Xiaochi Wei, Shuaiqiang Wang, Dawei Yin, Wenqi Fan, Hui Liu, and Jiliang Tang. Exploring the potential of large language models (LLMs) in learning on graphs. *ACM SIGKDD Explorations Newsletter*, 25(2):42–61, 2024b.

Bahare Fatemi, Jonathan Halcrow, and Bryan Perozzi. Talk like a graph: Encoding graphs for large language models. In *International Conference on Learning Representations*, 2024.

Xingbo Fu, Binchi Zhang, Yushun Dong, Chen Chen, and Jundong Li. Federated graph machine learning: A survey of concepts, techniques, and applications. *ACM SIGKDD Explorations Newsletter*, 24(2):32–47, 2022.

Zirui Guo, Lianghao Xia, Yanhua Yu, Yuling Wang, Zixuan Yang, Wei Wei, Liang Pang, Tat-Seng Chua, and Chao Huang. GraphEdit: Large language models for graph structure learning. *arXiv preprint arXiv:2402.15183*, 2024.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, pp. 1024–1034, 2017.

Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. Harnessing explanations: LLM-to-LM interpreter for enhanced text-attributed graph representation learning. In *The Twelfth International Conference on Learning Representations*, 2024.

Wei Huang, Chao Zhu, Yanyong Huang, Jia Liu, and Tianrui Li. A dual-branch federated graph learning model with global graph structure information. *Knowledge-Based Systems*, 328:114254, 2025.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lelio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothee Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Peter Kairouz, H. Brendan McMahan, Brendan Avent, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

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

Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, volume 2, pp. 429–450, 2020.

Rui Liu, Pengwei Xing, Zichao Deng, Anran Li, Cuntai Guan, and Han Yu. Federated graph neural networks: Overview, techniques, and challenges. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3):4279–4295, 2025.

H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1273–1282, 2017.

Bo Pan, Zheng Zhang, Yifei Zhang, Yuntong Hu, and Liang Zhao. Distilling large language models for text-attributed graph learning. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2024.

Yang Pei, Renxin Mao, Yang Liu, Chao-Shu Chen, Shifeng Xu, Feng Qiang, and Peng Zhang. Decentralized federated graph neural networks. In *Proceedings of the International Workshop on Federated and Transfer Learning for Data Sparsity and Confidentiality in Conjunction with IJCAI*, 2021.

Yue Tan, Yixin Liu, Guodong Long, Jing Jiang, Qinghua Lu, and Chengqi Zhang. Federated learning on non-IID graphs via structural knowledge sharing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 9953–9961, 2023.

Zihan Tan, Guancheng Wan, Wenke Huang, and Mang Ye. FedSSP: Federated graph learning with spectral knowledge and personalized preference. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

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

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

Binghui Wang, Ang Li, Meng Pang, Hai Li, and Yiran Chen. GraphFL: A federated learning framework for semi-supervised node classification on graphs. In *Proceedings of the IEEE International Conference on Data Mining*, pp. 498–507, 2022.

Lingren Wang, Wenxuan Tu, Jiaxin Wang, Xiong Wang, Jieren Cheng, and Jingxin Liu. FedIGL: Federated invariant graph learning for non-IID graphs. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. LLMRec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 806–815, 2024.

Ke Zhang, Carl Yang, Xiaoxiao Li, Lichao Sun, and Siu Ming Yiu. Subgraph federated learning with missing neighbor generation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 6671–6682, 2021.

Taolin Zhang, Chengyuan Mai, Yaomin Chang, Chuan Chen, Lin Shu, and Zibin Zheng. FedEgo: Privacy-preserving personalized federated graph learning with ego-graphs. *ACM Transactions on Knowledge Discovery from Data*, 18(2), 2024.

Jianan Zhao, Meng Qu, Chaozhuo Li, Hao Yan, Qian Liu, Rui Li, Xing Xie, and Jian Tang. Learning on large-scale text-attributed graphs via variational inference. In *International Conference on Learning Representations*, 2023.