Federated Graph Semantic and Structural Learning

Wenke Huang[1] , **Guancheng Wan**[1] , **Mang Ye**[1,2*] and **Bo Du**[1,2]
[1]School of Computer Science, Wuhan University, Wuhan, China
[2] Hubei Luojia Laboratory, Wuhan, China
{wenkehuang, guanchengwan, yemang, dubo}@whu.edu.cn

Abstract

Federated graph learning collaboratively learns a global graph neural network with distributed graphs, where the non-independent and identically distributed property is one of the major challenge. Most relative arts focus on traditional distributed tasks like images and voices, incapable of the graph structures. This paper firstly reveals that local client distortion is brought by both node-level semantics and graph-level structure. First, for node-level semantic, we find that contrasting nodes from distinct classes is beneficial to provide a well-performing discrimination. We pull the local node towards the global node of the same class and push them away from the global node of different classes. Second, we postulate that a well-structural graph neural network possesses similarity for neighbors due to the inherent adjacency relationships. However, aligning each node with adjacent nodes hinders discrimination due to the potential class inconsistency. We transform the adjacency relationships into the similarity distribution and leverage the global model to distill the relation knowledge into the local model, which preserves the structural information and discriminability of the local model. Empirical results on three graph datasets manifest the superiority of the proposed method over counterparts.

1 Introduction

Federated learning (*FL*) has shown considerable potential in collaborative machine learning across distributed devices without disclosing privacy [Konečnỳ *et al.*, 2016; Fang and Ye, 2022]. Although *FL* has attracted wide research interest and witnessed remarkable progress [McMahan *et al.*, 2017a; Zhan *et al.*, 2020], most of them focus on the tasks like images and voices on the basis of *CNN* and transformer [He *et al.*, 2016; Vaswani *et al.*, 2017]. However, many real-world applications generate structured graphical data (*e.g.*, knowledge graph, social network [Liu *et al.*, 2021]), consisting of vertices and edges [Panagopoulos *et al.*, 2021; Wang *et al.*, 2021], while *CNN* and transformer can not deal with them effectively due to the inability to capture the topological structure [Kipf and Welling, 2017; Zhu *et al.*, 2020]. For these graph applications, graph neural networks (*GNN*) have won praise for their impressive performance [Hamilton *et al.*, 2017; Shchur *et al.*, 2018] because they utilize both the independence of nodes and the unique structure of graphs to mine graph data. Therefore, for the purpose of handling the graph data across multiple participants with growing privacy concerns, Federated Graph Learning (*FGL*) has become a promising paradigm [Fu *et al.*, 2022].

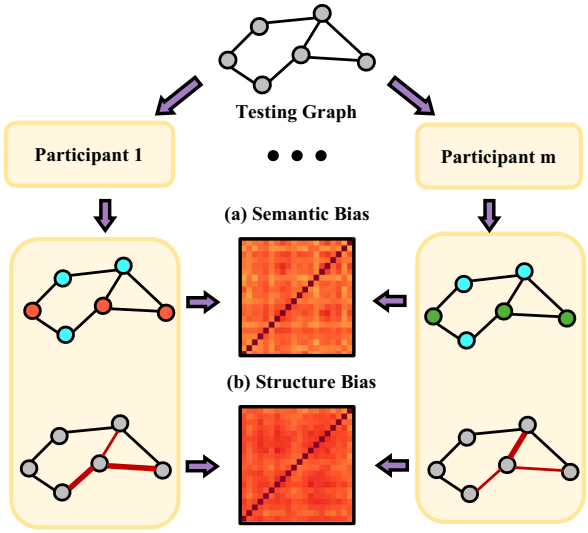


Figure 1: **Problem illustration.** We present the structure and semantic level similarities among clients. The deeper color suggests a more similar representation of node and graph across different participants, while the shallower mean dissimilarity. (a) Semantic bias: clients show the inconsistency predicted class of node. (b) Structure bias: clients hold distinct similarities among neighborhood nodes. In this work, we conduct semantics-level and structure-level calibration to achieve better federated graph learning performance.

Notably, data heterogeneity has become an inescapable problem in Federated Learning [Kairouz *et al.*, 2019; Huang *et al.*, 2022a]. Specifically, the data distribution among different parties presents non-IID (identically and independently distributed) property, which results in the divergence of local direction [Huang *et al.*, 2023; Huang *et al.*, 2022b]. Although existing methods have made efforts to restrain the lo-

*Corresponding authors.

cal model with respect to the global model, they mainly design for typical data heterogeneity without special consideration for graph structural bias. However, previous work has demonstrated the importance of exploiting structural knowledge [Liu *et al.*, 2023a; Liu *et al.*, 2022b]. In this paper, we preliminarily investigate the unique characteristics of graph heterogeneity and find that there exists node semantic and graph structural bias in *FGL* setting. In detail, we leverage the Centered Kernel Alignment (*CKA*) [Kornblith *et al.*, 2019] to measure the similarity between node representations given by different client model pairs. Especially, for structure heterogeneity, we utilize the Anonymous Walk Embedding (*AWE*) to generate a representation for the graph and exploit the Jensen-Shannon distance between pair of graphs to measure the discrepancy. We reveal that there exists severe divergence on both node-level semantic and graph-level structure among clients (Fig. 1). Therefore, a crucial problem for *FGL* is that *How to calibrate both node-level semantic and graph-level structure bias in Federated Graph Learning?*

For node-level semantic calibration, we argue that a well-discriminative graph neural network contrasts nodes from different classes to provide a clear decisional boundary. Inspired by the success of supervised contrastive learning [Khosla *et al.*, 2020; Liu *et al.*, 2023b], we naturally expect to conduct pull-push operations among different classes on local model to acquire a well-performing decisional ability. However, under federated learning, the local *GNN* model purely optimizes on private data and drifts towards a distinct local minimum, which means solely relying on the guidance signals provided by the local model is confusing and unreliable. Prior studies [Hu *et al.*, 2022], attempt to reweight local model during the FL aggregation process, but this sheds little light on identity node semantics bias caused by data heterogeneity. In this work, we investigate the node semantic knowledge and calibrate it during the local training process. We propose Federated Node Semantic Contrast (*FNSC*), which encourages the query local node representation to be close to **global** node embeddings within same class and pushes it away from **global** node embeddings of different classes.

Besides, for graph-level structural calibration, local clients normally possess a graph that is incomplete and is biased to depict the graph structure. Existing works normally focus on reconstructing the graph structure to handle the local structural bias. For example, FedStar [Tan *et al.*, 2023] decouples the structure information and encodes it in a personalized way, which brings extra model parameters for local updating. In this work, without more communication cost, we take a free ride to convert stiff graph structure reconstruction into structural relationship maintenance via the given global model during local training. We introduce Federated Graph Structure Distillation (*FGSD*). In detail, for each node, we leverage the global model to calculate the similarity of each node with its neighborhoods, based on the adjacency matrix. Then, we require the local model to generate the adjacent node similarity and mimic the global one, which leverages the global model to provide beneficial structural knowledge.

In a nutshell, we propose a novel Federated Graph Semantic and Structural Learning method (*FGSSL*). Our contributions are summarized as follows:

- We are the first in *FGL* to decouple the data heterogeneity setting to node semantic level and graph structural level bias respectively. From this perspective, we can ameliorate final degraded performance by calibrating the local training drift, which sheds good light on future research in solving the non-IID problem in *FGL* scenarios.
- We introduce a novel federated graph learning (*FGSSL*) frame for both node and graph-level calibration. The former Federated Node Semantic Contrast calibrates local node semantics with the assistance of the global model without compromising privacy. The latter Federated Graph Structure Distillation transforms the adjacency relationships from the global model to the local model, fully reinforcing the graph representation with aggregated relation.
- We conduct extensive experiments on benchmark datasets to verify that *FGSSL* achieves superior performance over related methods. Taking a free ride with the global model, it does not introduce additional communication rounds and shows stronger privacy since it does not require additional shared sensitive prior information.

2 Related Work

2.1 Federated Graph Learning

Federated graph learning (*FGL*) facilitates the distributed training of graph neural networks (*GNN*). Previous literature on *FGL* can be categorized into two types: inter-graph and intra-graph. Inter-graph *FGL* involves each participant possessing a set of graphs and collectively participating in federated learning (*FL*) to improve the modeling of local data or generate a generalizable model [Xie *et al.*, 2021]. In contrast, intra-graph *FGL* involves each participant owning only a subset of the entire graph and the objective is to address missing links [Zhang *et al.*, 2021] or discover communities [Baek *et al.*, 2022]. However, both of them are confronted with the non-IID issue which degrades the collaboratively learned model performance. Conventional methods solving the non-IID in *FL* field (*e.g.*, FedProx [Li *et al.*, 2020] and MOON [Li *et al.*, 2021]) meet the absence of design for *FGL* scenarios. Some preceding methods are dedicated to handling the non-IID problem for *FGL*. FedGCN [Hu *et al.*, 2022] tries to reweight local model parameters via an attention mechanism. FILT+ [Zhu *et al.*, 2021a] pulls the local model closer to the global model by minimizing the loss discrepancy between a local model and the global model. However, they focus on leveraging the issue from model respect and fail to effectively exploit the unique characteristics of the graph data. In this paper, we consider inter-graph *FGL* and deal with the non-IID via exploiting the graphic characteristics and decoupling into node-level semantic and graph-level structure calibration.

2.2 Contrastive Learning on Graphs

In recent years, contrastive learning has seen a resurgence of interest in the field of visual representation learning [He *et al.*, 2020; Chen *et al.*, 2020]. This success has spurred a wealth of research exploring the adaptation of contrastive learning to graph-like data for self-supervised methods [Zhu *et al.*, 2021b; Liu *et al.*, 2022a]. Traditional unsupervised methods on graph representation learning approaches [Grover

and Leskovec, 2016; Perozzi *et al.*, 2014], adhere to a contrastive structure derived from the skip-gram model. The graph autoencoder (GAE) [Kipf and Welling, 2016] is a self-supervised learning technique that aims to reconstruct the graph structure while The MVGRL [Hassani and Khasahmadi, 2020] intends to do node diffusion and compare node representation to augmented graph representation in order to learn both node-level and graph-level representation. Similar to SimCLR [Chen *et al.*, 2020], GRACE [Zhu *et al.*, 2020] constructs two augmented views of a graph by randomly perturbing nodes and edges, and subsequently learns node representations by pushing apart representations of every other node while bringing together representations of the same node in the two different augmented graphs within the same network. Apart from self-supervised tasks, SupCon [Khosla *et al.*, 2020] firstly extend the self-supervised batch contrastive approach to the fully-supervised setting. In this work, we examine the contrastive method in distributed systems and conduct a inter-view based contrast between the global and local models respectively. Moreover, we consider the supervised contrast that leveraging the label as a signal to choose positive samples for calibrating the node embedding to be more similar to the global node embedding.

2.3 Knowledge Distillation

Knowledge Distillation (*KD*) [Hinton *et al.*, 2015] is a technique that has been extensively studied and applied in various areas of machine learning, including image classification, natural language processing, and graph representation learning. The key aspect of *KD* is transferring knowledge from a complex and powerful teacher model to a more limited student model. In many works, knowledge distillation is typically used to train a smaller student network under the guidance of a larger teacher network with minimal to no performance degradation [Wang and Yoon, 2021]. In practice, knowledge distillation forces the feature or logit output of the student network to be similar to that of the teacher network. Researchers have attempted to improve knowledge distillation methods by introducing new techniques such as model distillation [Mullapudi *et al.*, 2019], feature distillation [Romero *et al.*, 2015], and relation distillation [Park *et al.*, 2019]. In this work, we focus on ameliorating the heterogeneity of graph structure by adapting relation-based *KD* techniques for the *FGL* domain. We first transform the adjacency relationships into similarity distribution from global view, then distill them into the local model. In this way, we leverage aggregated contextual neighborhood information from global view and calibrate the drift caused by graph structure from the locally biased data.

3 Methodology

3.1 Preliminaries

Graph Neural Newrok. Graph neural networks (*GNN*), *e.g.*, graph convolutional networks (*GCN*) [Kipf and Welling, 2017] and Graph Attention Networks (*GAT*) ([Veličković *et al.*, 2017]), improved the state-of-the-art in informative graph data with their elegant yet powerful designs. In general, given the structure and feature information of a graph $\mathcal{G} = (V, A, X)$, where V, A, X denote nodes, adjacency matrix and node feature respectively, *GNN* targets to learn the representations of graphs, such as the node embedding $h_i \in \mathbb{R}^d$. A *GNN* typically involves two steps: the processes of message propagation and neighborhood aggregation. In this process, each node in the graph iteratively collects information from its neighbors with its own information in order to update and refine its representation. Generally, an L-layer *GNN* can be formulated as

$$h_i^{(l+1)} = \sigma(h_i^{(l)}, AGG(\{h_j^{(l)}; j \in A_i\})), \forall l \in [L], \quad (1)$$

where $h_i^{(l)}$ denotes the representation of node v at the l^{th} layer, and $h_i^{(0)} = v_i$ represents the node feature. A_i is defined as the neighbors of node v_i, $AGG(\cdot)$ is a aggregation function that can vary for different *GNN* variants, and σ means a activation function.

After L message-passing layers, the final node embedding h_i is passed to a project head F to obtain logits:

$$z_i = F(h_i). \quad (2)$$

In this paper, we examine proposed *FGSSL* in node-level tasks (*e.g.*, node classification), and F is defined as the classifier head. Specially, we utilize L-1 layers as *GNN* feature extractor and the L layer as F.

Centralized Aggregation

In vanilla FL setting there is always a central server with M clients, the m-th client owns a private dataset D^m and $|D|$ is the total size of samples over all clients. FedAvg [McMahan *et al.*, 2017b] is a foundational algorithm in the field of federated learning, which serves as a starting point for the design of more advanced *FL* frameworks. It operates by aggregating the updated model parameters from individual clients and redistributing average of these parameters back to all clients:

$$\theta \leftarrow \sum_{m=1}^{M} \frac{|D^m|}{|D|} \theta^m. \quad (3)$$

In this study, we utilize the Federated Learning (*FL*) framework to enable collaborative learning on isolated graphs among multiple data owners, without the need to share raw graph data. By doing so, we aim to obtain a global node classifier. Specifically, when model parameters are set to θ for the Graph Neural Network (*GNN*) encoder and classifier F, we formalize the global objective:

$$\arg\min \frac{1}{M} \sum_{m}^{M} \mathcal{L}^m(\theta^m; D^m). \quad (4)$$

Normally, the loss function $\mathcal{L}^m$ in Eq. (5) is cross-entropy loss as each node which is optimized with softmax operation:

$$\mathcal{L}_i^{CE} = -\mathbf{1}_{c_i} \log(\texttt{softmax}(\boldsymbol{z_i})), \quad (5)$$

where $\mathbf{1}_{c_i}$ denotes the one-hot encoding of the label c_i.

3.2 Motivation

Commonly, federated graph learning aims at training a shared global *GNN* model, where clients have their own graphs and

do not expose private data. In real-world applications, heterogeneous data distribution exists among clients. Therefore, clients present divergent optimization directions, which impair the performance of the global *GNN* model. We also show that this client divergence manifests in node-level semantics and graph-level structure aspects. We leverage the pairwise Centered Kernel Alignment (*CKA*) [Kornblith *et al.*, 2019] and calculate the similarity between arbitrary *GNN* models on the same input testing samples. *CKA* generates the similarity score ranging from 0 (not at all similar) to 1 (identical). We select 20 clients and train the local *GNN* model for 100 epochs, simultaneously taking the node output from different models as node representation. As shown in Fig. 1, considering both node semantics and graph structure calibration into account is beneficial to learning a better shared *GNN* model.

3.3 Proposed Method

Federated Node Semantic Contrast. Generally, the goal of node classification is to identify all samples. Thus, the *GNN* module should maintain the discernible patterns. Inspired by the success of supervised contrastive learning, we naturally expect to contrast the node features of different classes. For the local model, we pull the node feature vectors closer to the positive samples from the same semantics and push them far away from negativeness with distinct classes. Specifically, for the node v_i, its embedding h_i^m generated by local *GNN* encoder $G^m(\cdot)$ with its ground truth c_i, the positive samples are other nodes belonging to the same class c_i, while the negatives are the nodes from the different classes $\mathcal{C}\backslash c_i$. Our supervised, local node-wise contrastive loss is defined as:

$$\mathcal{L}_i^{CON} = \frac{-1}{|\boldsymbol{P}_i|} \sum_{p \in \boldsymbol{P}_i} \log \frac{\varphi(h_i^m, h_p^m, \tau)}{\varphi(h_i^m, h_p^m, \tau) + \sum_{k \in \boldsymbol{K}_i} \varphi(h_i^m, h_k^m, \tau)}, \tag{6}$$

where $\boldsymbol{P}_i$ and $\boldsymbol{K}_i$ denote the collections of the positive and negative samples sets for the node v_i. We define the τ as a contrastive hyper-parameter and φ is formulated as:

$$\varphi(h_i, h_j, \tau) = exp(\frac{h_i \cdot h_j}{||h_i||\ ||h_j||}/\tau). \tag{7}$$

However, it is widely known that private models present drift from the ideal global optima. Thus, naively leveraging the private model to provide the positive and negative sets would further skew the local optimization direction. In our work, we argue that the shared global model aggregates knowledge from multiple parties and presents less bias than the local model. In this paper, we propose Federated Node Semantic Contrast (*FNSC*), which leverages the global model to provide positive and negative cluster representations for each local node embedding. We further reformulate the aforementioned supervised node contrastive learning as follows:

$$\mathcal{L}_i^{FNSC} = \frac{-1}{|\boldsymbol{P}_i|} \sum_{p \in \boldsymbol{P}_i} \log \frac{\varphi(h_i^m, h_p^g, \tau)}{\varphi(h_i^m, h_p^g, \tau) + \sum_{k \in \boldsymbol{K}_i} \varphi(h_i^m, h_k^g, \tau)}, \tag{8}$$

where h^g denotes the the node embedding generated by the *GNN* encoder $G^g(\cdot)$ Moreover, given the node embedding h_i^m generated by local *GNN* encoder $G^m(\cdot)$, we pull the node v_i from local view and its pairwise one h_i^g in global view together, simultaneously pull it and nodes from global view with the same class c_i together.

Notably, the recent success of contrastive learning in image or video processing is largely due to carefully designed image augmentations [Ye *et al.*, 2022; Ye *et al.*, 2019]. These augmentations allow the model to explore a wider range of underlying semantic information and obtain better performance. In this section, we adopt a similar strategy for graph data by using an augmentation module, denoted by $Aug(\cdot)$, to generate two different views of the graph. Prior research has produced various methods for graph augmentation, which can be divided into two categories: topology (structure) transformation and feature transformation (*e.g.*, Edge Removing and Feature Masking) [Zhu *et al.*, 2021b; Zhu *et al.*, 2020]. In order to enforce local clients to acquire a well-discriminative ability, we leverage both augmentations in our augmentation modules. Furthermore, we propose an **asymmetric** design for the contrast process, which utilizes stronger augmentations for the local *GNN* and weaker augmentations for the global *GNN*, given by $\widetilde{\mathcal{G}}_1 = Aug_s(\mathcal{G})$ for strong $Aug(\cdot)$ and $\widetilde{\mathcal{G}}_2 = Aug_w(\mathcal{G})$ for weak $Aug(\cdot)$. This would give local clients great strength to optimize towards the global direction, meanwhile, the global model can provide stable contextual semantic information to local training process. We further demonstrate the effectiveness of this asymmetric augmentation strategy in Tab. 3.

Federated Graph Structure Distillation. For graph-level calibration, it is normally assumed that adjacent nodes will share similar representations. However, under federated learning, each client fails to effectively depict this relationship because local data is normally incomplete. The straightforward solution is to directly align the query local node feature with the neighborhood nodes from the global model. However, it could potentially disrupt the discriminability because neighboring nodes probably belong to different classes. Motivated by similarity knowledge distillation [Fang *et al.*, 2021; Tejankar *et al.*, 2021], we propose Federated Graph Structure Distillation (*FGSD*) to overcome semantic inconsistency of adjacent nodes, which maintains graphic structure knowledge via the support of the global model. We measure the similarity of the query node with neighboring nodes from the global model output and then optimize the local network to mimic the similarity distribution from the global view. Specifically, for the node v_i, z_i is the logit output given by the node classifier F, we denote the A_i as the neighborhood node set and define the $S^g(v_i, A_i)$ as the similarity of the selected node vector with adjacent nodes computed by the **global** model:

$$\begin{aligned} S^g(v_i, A_i) &= [S_1^g, \ldots, S_{|A_i|}^g], \\ S_j^g &= \frac{\exp\left(\left(z_i^g \cdot z_j^{g\,T}\right)/\omega\right)}{\sum_{j \in A_i} \exp\left(\left(z_i^g \cdot z_j^{g\,T}\right)\right)/\omega)}, \end{aligned} \tag{9}$$

where ω is the distillation hyper-parameter, $(\ )^T$ means transpose operation. Then, we measure similarity distribution

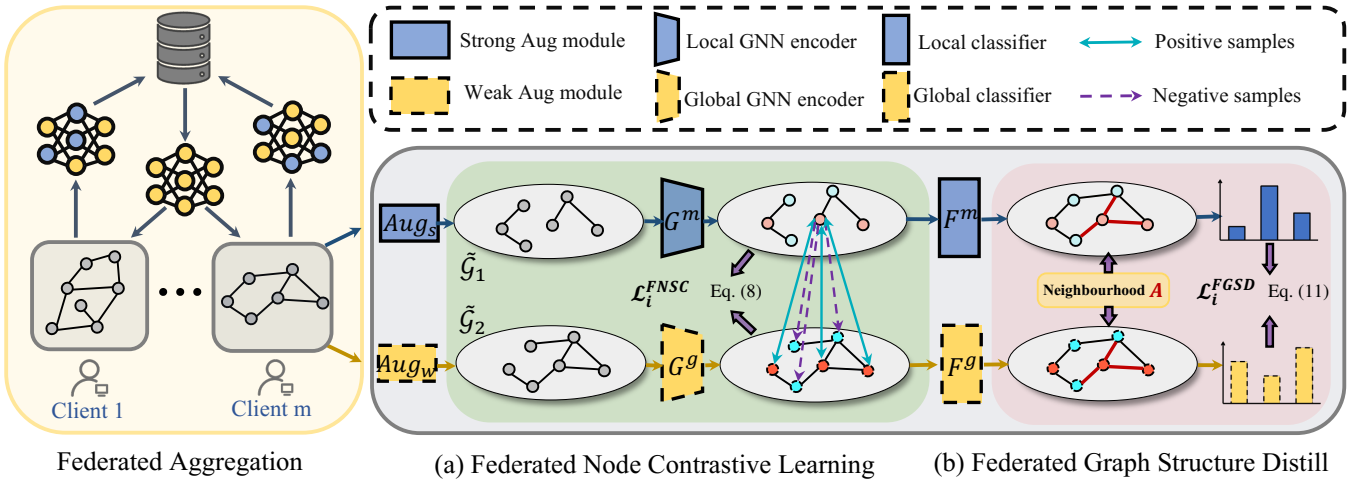


Figure 2: **Architecture illustration** of Federated Graph Semantic and Structural Learning (*FGSSL*). The left yellow box corresponds to the federated aggregation scheme (*e.g.* FedAvg), while the right grey box suggests the local training process. *FGSSL* includes two components: (a) Federated Node Semantic Contrast and (b) Federated Graph Structure Distillation. Best viewed in color. Zoom in for details.

from the m local model, $S^m(v_i, A_i)$, which is formed by:

$$S^m(v_i, A_i) = [S^m_1, \ldots, S^m_{|A_i|}],$$
$$S^m_j = \frac{\exp\left((z^m_i \cdot z^{mT}_j)/\omega\right)}{\sum_{j \in A_i} \exp\left((z^m_i \cdot z^{mT}_j)\right)/\omega)}. \quad (10)$$

The *FGSD* (Federated Graph Structure Distillation) loss is calculated as the following:

$$\mathcal{L}^{FGSD}_i = S^g(v_i, A_i) \log \frac{S^g(v_i, A_i)}{S^m(v_i, A_i)}. \quad (11)$$

Finally, the overall objective to be maximized is then formalized as the average across all nodes of the accumulation of the losses discussed above and is defined by:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \left(\mathcal{L}^{CE}_i + \lambda_C \mathcal{L}^{FNSC}_i + \lambda_D \mathcal{L}^{FGSD}_i \right). \quad (12)$$

To sum up, Federated Graph Semantic and Structural Learning *FGSSL* leverage the global model to simultaneously calibrate local model from the node-level semantics and graph-level structure, which effectively handles the heterogeneous graph data and learns a well-performing global *GNN* model. We further illustrate the *FGSSL* in the Algorithm 1.

4 Experiments

4.1 Experimental Setup

In this paper, we perform experiments on node-level tasks defined on graph data: we choose node classification to confirm the efficacy of *FGSSL* in various testing environments.

Datasets. For node classification, our experiments are conducted on three benchmark datasets for the citation networks:

- **Cora** [McCallum *et al.*, 2000] dataset consists of 2708 scientific publications classified into one of seven classes. There are 5429 edges in the network of citations. 1433 distinct words make up the dictionary.
- **Citeseer** [Giles *et al.*, 1998] dataset consists of 3312 scientific publications classified into one of six classes and 4732 edges. The dictionary contains 3703 unique words.
- **Pubmed** [Sen *et al.*, 2008] dataset consists of 19717 scientific papers on diabetes that have been categorized into one of three categories in the PubMed database. The citation network has 44338 edges in it. A word vector from a dictionary with 500 unique terms that is TF/IDF weighted is used to describe each publication in the dataset.

Network Structure. Since the *GAT* [Veličković *et al.*, 2017] is a powerful and widely used benchmark network in graph representing learning, we realize two layers *GAT* with parameter θ, decoupling it into feature extractor $G(\cdot)$ and unified classifier $F(\cdot)$. The hidden dimensions are 128 for all datasets, and classifier F maps the embedding from 128 dimensions to 7,6,3 dimensions, which is the number of classification classes for Cora, Citeseer, and Pubmed respectively.

Graph Augmentation Strategy. Generating views is a key component of contrastive learning methods. In the graph domain, different views of a graph provide different contexts for each node. We follow augmentation mentioned in [Zhu *et al.*, 2021b], [Zhu *et al.*, 2020] to construct a contrastive learning scheme. In *FGSSL*, we leverage two methods for new graph view generation, removing edges for topology and masking features for node attributes.

- **Removing edges (RE)**. It randomly removes a portion of edges in the original graph.
- **Masking node features (MF)**. It randomly masks a fraction of dimensions with zeros in node features.

Implement Details. We utilize the community detection algorithm: Louvain, to simulate the subgraph systems. To stimulate the non-iid scene, this algorithm partitions the graph into multiple clusters and then assigns them to distributed clients. To conduct the experiments uniformly and fairly, we split the nodes into train/valid/test sets, where the ratio is 60% : 20% : 20% . As for all networks, we use *SGD* [Robbins and Monro, 1951] as the selected optimizer with momentum 0.9 and weight decay $5e-4$. The communication round is 200 and the local training epoch is 4 for all datasets. The metric used in our experiments is the node classification accuracy on the testing nodes and we report the averaged accuracy and the

Algorithm 1: The *FGSSL* Framework

Input: communication rounds T, local epochs E, participant scale M, m^{th} client private graph data $\mathcal{G}^m(V, A, X; Y)$, private model θ^m, temperature τ, distillation parameter ω, loss weight λ_C and λ_D, learning rate η

Output: The final global model θ_t

for $t = 1, 2, ..., T$ **do**
  Participant Side;
  for $m = 1, 2, ..., M$ **in parallel do**
    send the global model θ_t to m-th client
    $\theta_t^m \leftarrow$ **LocalUpdating**(θ_t, m)
  end
  Server Side;
  $\theta_{t+1} \leftarrow \sum_{m=1}^{M} \frac{|D^m|}{|D|} \theta_t^m$
end
return θ_t

LocalUpdating(θ_t, m):

Initialize $G^g(\cdot), F^g(\cdot) \leftarrow \theta_t$

Initialize $G^m(\cdot), F^m(\cdot) \leftarrow \theta_t$

Freeze $G^g(\cdot), F^g$

for $e = 1, 2, ..., E$ **do**
  $Z = \theta^m(X)$
  $L^{CE} \leftarrow CE(Z, Y)$ in Eq. (5)
  $\widetilde{\mathcal{G}}_1, \widetilde{\mathcal{G}}_2 \leftarrow Aug_s(\mathcal{G}), Aug_w(\mathcal{G})$
  $H^m, H^g \leftarrow G^m(\widetilde{\mathcal{G}}_1), G^g(\widetilde{\mathcal{G}}_2)$
  $L^{FNSC} \leftarrow (H^m, H^g)$ through Eq. (8)
  $Z^m, Z^g \leftarrow F^m(H^m), F^g(H^g)$
  $S^g(V, A) \leftarrow (Z^g)$ by Eq. (9)
  $S^m(V, A) \leftarrow (Z^m)$ by Eq. (10)
  $L^{FGSD} \leftarrow (S^m(V, A), S^g(V, A))$ through Eq. (12)
  $\mathcal{L} = \mathcal{L}^{CE} + \lambda_C \mathcal{L}^{FNSC} + \lambda_D \mathcal{L}^{FGSD}$
  $\theta^m \leftarrow \theta^m - \eta \nabla \mathcal{L}$
end
return θ^m

standard deviation over several random repetitions.

Counterparts. (1) **Local** each client train their model locally, (2) **Global** the server leverage the complete graph for training. For rigorous evaluation, we compare our *FGSSL* against popular federated strategies in *FGL* setting. (3) **FedAvg** (AISTATS'17 [McMahan *et al.*, 2017b]), (4) **FedProx** (MLSys'21 [Li *et al.*, 2020]), (5) **FedOpt** (ICLR'21 [Reddi *et al.*, 2021]), (6) **FedSage** (NeurIPS'21 [Zhang *et al.*, 2021]).

4.2 Experimental Results

Performance Comparison. The results of federated node classification for various methods under three non-IID settings are presented in Tab. 1. These results indicate that *FGSSL* outperforms all other baselines and demonstrates a significant and consistent improvement compared to the conventional FedAvg algorithm in the *FGL* setting. Additionally, personalized *FL* algorithms such as FedProx and FedOpt demonstrate better performance than vanilla aggregation by utilizing a universal solution to the non-IID problem. Specialized methods in the *FGL* field such as FedSage also perform better than common baselines, which is achieved through the simultaneous training of generative models for predicting missing links.

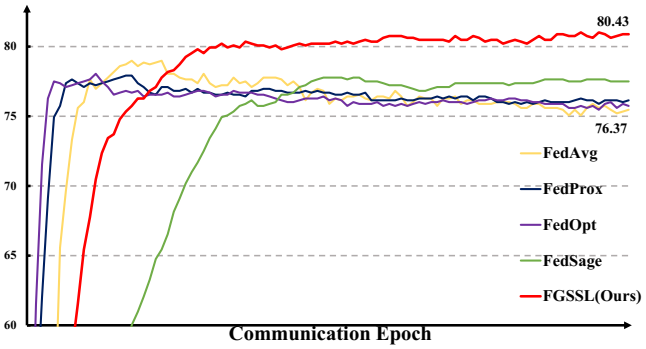
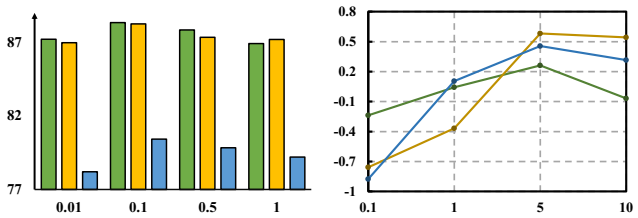


Figure 3: **Visualization of training curves** of the average test accuracy with Communication Epochs 200 with Citeseer dataset. Please see Sec. 4.2 for details.



(a) Contrastive parameter τ (b) Distillation parameter ω

Figure 4: **Analysis on hyper-parameter in *FGSSL***. Node classification results on three datasets under different τ and ω values with M = 5, in which green represents Cora, yellow represents Pubmend, and blue represents Citeseer. Refer to Sec. 4.3 for details.

Convergence Analysis. Fig. 3 shows curves of the average test accuracy during the training process across five random runs conducted on the Citeseer datasets. It can be observed that *FGSSL* dominates the other methods in non-IID setting on the average test accuracy and achieves a stable convergence.

4.3 Ablation Study

Effects of Key Components Mechanism. To better understand the impact of specific design components on the overall performance of *FGSSL*, we conducted an ablation study in which we varied these components. For the variant without *FNSC* and *FGSD*, we utilize the vanilla *FGL* setting with 2-layer *GAT*. As shown in Tab. 2, by exploiting both components, the best performance is achieved in all three graph datasets. It also suggests that *FNSC* plays a more crucial role than *FGSD*, which means the calibration in node semantics is stronger than the calibration in graph structure, and feature heterogeneity is more serious than graph heterogeneity in non-IID setting. Moreover, the contribution made by *FGSD* is still not negligible and can benefit the learning process.

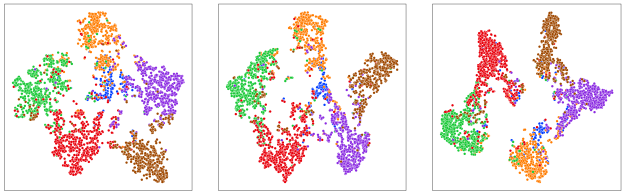
Hyper-parameter study. We compare the downstream task performance under different τ and ω values with five clients. Results are shown in Tab. 1, where Fig. 4(a) shows results when ω is fixed at 5, and Fig. 4(b) shows results under $\tau = 0.1$. It indicates that choosing τ can affect the strength of the contrastive method, where a smaller temperature benefits

Methods	Cora			Citeseer			Pubmed		
	$M=5$	$M=7$	$M=10$	$M=5$	$M=7$	$M=10$	$M=5$	$M=7$	$M=10$
Global	87.78±1.34			76.91±1.02			88.38±0.33		
Local	61.54±0.83	45.32±1.52	32.42±2.81	73.85±1.20	62.87±2.45	48.91±2.34	83.81±0.69	72.34±0.79	59.19±1.31
FedAvg	86.63±0.35	86.21±0.21	86.01±0.17	76.37±0.43	76.57±0.46	75.92±0.21	85.29±0.83	84.27±0.29	84.57±0.29
FedProx	86.60±0.59	86.27±0.12	86.22±0.25	77.15±0.45	77.28±0.78	76.87±0.80	85.21±0.24	84.01±0.59	84.98±0.65
FedOpt	86.11±0.24	85.89±0.43	85.20±0.93	76.96±0.34	76.82±0.04	76.71±0.19	84.39±0.42	84.10±0.19	83.91±0.20
FedSage	86.86±0.15	86.59±0.23	86.32±0.37	77.91±0.59	77.82±0.13	77.30±0.71	87.75±0.23	87.51±0.20	87.49±0.09
FGSSL	**88.34**±0.34	**88.56**±0.43	**88.01**±0.26	**80.43**±0.23	**80.21**±0.11	**80.01**±0.09	**88.25**±0.60	**87.75**±0.41	**87.60**±0.53
	↑ 1.71	**↑ 2.35**	**↑ 2.00**	**↑ 4.06**	**↑ 3.64**	**↑ 4.09**	**↑ 2.96**	**↑ 3.48**	**↑ 2.73**

Table 1: **Comparison with the state-of-the-art methods** on Cora, Citeseer and Pubmed datasets. The best result is bolded. ↑ means improved accuracy compared with FedAvg. ± presents the standard deviation. Please see details in Sec. 4.2.

FNSC	*FGSD*	Cora			Citeseer		
		$M=5$	$M=7$	$M=10$	$M=5$	$M=7$	$M=10$
✗	✗	86.63	86.21	86.01	76.37	76.57	75.92
✗	✓	86.86	86.32	86.51	77.91	77.53	76.42
✓	✗	88.01	88.23	87.84	79.89	79.43	79.12
✓	✓	**88.34**	**88.56**	**88.01**	**80.43**	**80.21**	**80.01**

Table 2: **Ablation study of key components** of our method in Cora and Citeseer datasets with clients 5/7/10. See Sec. 4.3 for details.



(a) FedAvg (b) FedSage (c) FGSSL

Figure 5: **Visualization of classification result.** The figure number corresponds to the method on the Citeseer dataset with $m = 5$. Logits are colored based on class labels.

training more than higher ones, but extremely low temperatures (0.01) are harder to train due to numerical instability. Across different datasets, the optimal τ is constantly around 0.1. For choosing an appropriate τ in (Eq. (9) and Eq. (10).), we find that the performance is not influenced much unless ω is set to extreme values like 0.1.

Discussion on Augmentation Strategies. As demonstrated in Tab. 3, different augmentation strategies were implemented within the augmentation module of proposed method. The experimental results indicate that utilizing two levels of augmentation improves performance. Specifically, on the one hand, using double-weak augmentation strategies did not result in a significant improvement when compared to baseline methods. On the other hand, double-strong augmentation strategies led to improved results as they allowed for exploration of rich semantic information through the supervised contrastive method. Additionally, the combination of strong and weak augmentation strategies at local and global levels, respectively, resulted in the highest overall performance, in accordance with our descriptions of them in Sec. 3.3.

Local	Global	Cora			Citeseer		
		$M=5$	$M=7$	$M=10$	$M=5$	$M=7$	$M=10$
weak	weak	87.24	87.10	86.99	77.72	77.45	77.30
weak	strong	86.86	86.68	86.48	77.22	77.09	76.33
strong	strong	87.91	87.93	87.52	79.59	79.12	78.81
strong	weak	**88.01**	**88.23**	**87.84**	**79.89**	**79.43**	**79.12**

Table 3: **Analysis on augmentation strategies :** Effect of using weak or strong augmentations for two datasets trained on the sole *FNSC* component with 200 epochs. See Sec. 4.3 for details.

5 Conclusion

In this paper, we propose a novel federated graph learning framework, namely *FGSSL*, that mitigates the non-IID issues via appropriately calibrating the heterogeneity both on the node-level semantic and graph-level structure. We develop two key components to solve the problems respectively. On the one hand, we leverage the contrastive-based method to correct the drift node semantics from the global ones that have Identical semantic information and achieve a high level of semantic discrimination at node level. On the other hand, we consider transforming adjacency relationships into a similarity distribution and utilizing a global model to distill this information into the local model, which maintains the structural information and corrects the structure heterogeneity. Experimental results illustrate that *FGSSL* consistently outperforms the state-of-the-art methods in federated graph scenarios.

Acknowledgments

This work is partially supported by National Natural Science Foundation of China under Grant (62176188, 62225113), the Key Research and Development Program of Hubei Province (2021BAA187), Zhejiang lab (NO.2022NF0AB01), CCF-Huawei Populus Grove Fund (CCF-HuaweiTC2022003), the Special Fund of Hubei Luojia Laboratory (220100015) and the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant (2019AEA170).

Contribution Statement

Wenke Huang and Guancheng Wan contributed equally to this work.

References

[Baek *et al.*, 2022] Jinheon Baek, Wonyong Jeong, Jiongdao Jin, Jaehong Yoon, and Sung Ju Hwang. Personalized subgraph federated learning. *arXiv preprint arXiv:2206.10206*, 2022.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020.

[Fang and Ye, 2022] Xiuwen Fang and Mang Ye. Robust federated learning with noisy and heterogeneous clients. In *CVPR*, 2022.

[Fang *et al.*, 2021] Zhiyuan Fang, Jianfeng Wang, Lijuan Wang, Lei Zhang, Yezhou Yang, and Zicheng Liu. Seed: Self-supervised distillation for visual representation. In *ICLR*, 2021.

[Fu *et al.*, 2022] Xingbo Fu, Binchi Zhang, Yushun Dong, Chen Chen, and Jundong Li. Federated graph machine learning: A survey of concepts, techniques, and applications. *arXiv preprint arXiv:2207.11812*, 2022.

[Giles *et al.*, 1998] C Lee Giles, Kurt D Bollacker, and Steve Lawrence. Citeseer: An automatic citation indexing system. In *Proceedings of the third ACM conference on Digital libraries*, pages 89–98, 1998.

[Grover and Leskovec, 2016] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *ACM SIGKDD*, pages 855–864, 2016.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *NeurIPS*, 30, 2017.

[Hassani and Khasahmadi, 2020] Kaveh Hassani and Amir Hosein Khasahmadi. Contrastive multi-view representation learning on graphs. In *ICML*, pages 4116–4126, 2020.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[He *et al.*, 2020] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[Hu *et al.*, 2022] Kai Hu, Jiasheng Wu, Yaogen Li, Meixia Lu, Liguo Weng, and Min Xia. Fedgcn: Federated learning-based graph convolutional networks for non-euclidean spatial data. *Mathematics*, 10(6):1000, 2022.

[Huang *et al.*, 2022a] Wenke Huang, Mang Ye, and Bo Du. Learn from others and be yourself in heterogeneous federated learning. In *CVPR*, 2022.

[Huang *et al.*, 2022b] Wenke Huang, Mang Ye, Bo Du, and Xiang Gao. Few-shot model agnostic federated learning. In *ACM MM*, pages 7309–7316, 2022.

[Huang *et al.*, 2023] Wenke Huang, Mang Ye, Zekun Shi, He Li, and Bo Du. Rethinking federated learning with domain shift: A prototype view. In *CVPR*, 2023.

[Kairouz *et al.*, 2019] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[Khosla *et al.*, 2020] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. volume 33, pages 18661–18673, 2020.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[Konečnỳ *et al.*, 2016] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[Kornblith *et al.*, 2019] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *ICML*, pages 3519–3529, 2019.

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

[Li *et al.*, 2021] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *CVPR*, pages 10713–10722, 2021.

[Liu *et al.*, 2021] Yixin Liu, Zhao Li, Shirui Pan, Chen Gong, Chuan Zhou, and George Karypis. Anomaly detection on attributed networks via contrastive self-supervised learning. *IEEE TNNLS*, 2021.

[Liu *et al.*, 2022a] Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and Philip Yu. Graph self-supervised learning: A survey. *IEEE TKDE*, 2022.

[Liu *et al.*, 2022b] Yixin Liu, Yu Zheng, Daokun Zhang, Hongxu Chen, Hao Peng, and Shirui Pan. Towards unsupervised deep graph structure learning. In *Proceedings of the ACM Web Conference 2022*, pages 1392–1403, 2022.

[Liu *et al.*, 2023a] Yixin Liu, Kaize Ding, Huan Liu, and Shirui Pan. Good-d: On unsupervised graph out-of-distribution detection. In *WSDM*, 2023.

[Liu *et al.*, 2023b] Yixin Liu, Yizhen Zheng, Daokun Zhang, Vincent Lee, and Shirui Pan. Beyond smoothing: Unsupervised graph representation learning with edge heterophily discriminating. In *AAAI*, 2023.

[McCallum *et al.*, 2000] Andrew Kachites McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

[McMahan *et al.*, 2017a] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[McMahan *et al.*, 2017b] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *AISTATS*, pages 1273–1282, 2017.

[Mullapudi *et al.*, 2019] Ravi Teja Mullapudi, Steven Chen, Keyi Zhang, Deva Ramanan, and Kayvon Fatahalian. Online model distillation for efficient video inference. In *ECCV*, pages 3573–3582, 2019.

[Panagopoulos *et al.*, 2021] George Panagopoulos, Giannis Nikolentzos, and Michalis Vazirgiannis. Transfer graph neural networks for pandemic forecasting. In *AAAI*, pages 4838–4845, 2021.

[Park *et al.*, 2019] Wonpyo Park, Dongju Kim, Yan Lu, and Minsu Cho. Relational knowledge distillation. In *CVPR*, pages 3967–3976, 2019.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *ACM SIGKDD*, pages 701–710, 2014.

[Reddi *et al.*, 2021] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečnỳ, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. In *ICLR*, 2021.

[Robbins and Monro, 1951] Herbert Robbins and Sutton Monro. A stochastic approximation method. *AoMS*, pages 400–407, 1951.

[Romero *et al.*, 2015] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. In *ICLR*, 2015.

[Sen *et al.*, 2008] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93–93, 2008.

[Shchur *et al.*, 2018] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018.

[Tan *et al.*, 2023] Yue Tan, Yixin Liu, Guodong Long, Jing Jiang, Qinghua Lu, and Chengqi Zhang. Federated learning on non-iid graphs via structural knowledge sharing. In *AAAI*, 2023.

[Tejankar *et al.*, 2021] Ajinkya Tejankar, Soroush Abbasi Koohpayegani, Vipin Pillai, Paolo Favaro, and Hamed Pirsiavash. Isd: Self-supervised learning by iterative similarity distillation. In *ICCV*, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017.

[Veličković *et al.*, 2017] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[Wang and Yoon, 2021] Lin Wang and Kuk-Jin Yoon. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE TPAMI*, 2021.

[Wang *et al.*, 2021] Zifeng Wang, Rui Wen, Xi Chen, Shilei Cao, Shao-Lun Huang, Buyue Qian, and Yefeng Zheng. Online disease diagnosis with inductive heterogeneous graph convolutional networks. In *WWW*, pages 3349–3358, 2021.

[Xie *et al.*, 2021] Han Xie, Jing Ma, Li Xiong, and Carl Yang. Federated graph classification over non-iid graphs. *NeurIPS*, 34:18839–18852, 2021.

[Ye *et al.*, 2019] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, pages 6210–6219, 2019.

[Ye *et al.*, 2022] Mang Ye, Jianbing Shen, Xu Zhang, Pong C. Yuen, and Shih-Fu Chang. Augmentation invariant and instance spreading feature for softmax embedding. *IEEE TPAMI*, 44(2):924–939, 2022.

[Zhan *et al.*, 2020] Yufeng Zhan, Peng Li, Zhihao Qu, Deze Zeng, and Song Guo. A learning-based incentive mechanism for federated learning. *IEEE Internet of Things Journal*, 7(7):6360–6368, 2020.

[Zhang *et al.*, 2021] Ke Zhang, Carl Yang, Xiaoxiao Li, Lichao Sun, and Siu Ming Yiu. Subgraph federated learning with missing neighbor generation. *NeurIPS*, 34:6671–6682, 2021.

[Zhu *et al.*, 2020] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. Deep graph contrastive representation learning. In *ICML*, 2020.

[Zhu *et al.*, 2021a] Wei Zhu, Andrew White, and Jiebo Luo. Federated learning of molecular properties in a heterogeneous setting. *arXiv preprint arXiv:2109.07258*, 2021.

[Zhu *et al.*, 2021b] Yanqiao Zhu, Yichen Xu, Qiang Liu, and Shu Wu. An empirical study of graph contrastive learning. *arXiv preprint arXiv:2109.01116*, 2021.