# Modeling Inter-Intra Heterogeneity for Graph Federated Learning

**Wentao Yu**[1], **Shuo Chen**[2*], **Yongxin Tong**[3], **Tianlong Gu**[4], **Chen Gong**[5*]
[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[2]Center for Advanced Intelligence Project, RIKEN, Japan
[3]State Key Laboratory of Complex & Critical Software Environment, Beihang University, China
[4]Engineering Research Center of Trustworthy AI (Ministry of Education), Jinan University, China
[5]Department of Automation, Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China
wentao.yu@njust.edu.cn, shuo.chen.ya@riken.jp, yxtong@buaa.edu.cn, gutianlong@jnu.edu.cn, chen.gong@sjtu.edu.cn

## Abstract

Heterogeneity is a fundamental and challenging issue in federated learning, especially for the graph data due to the complex relationships among the graph nodes. To deal with the heterogeneity, lots of existing methods perform the weighted federation based on their calculated similarities between pairwise clients (*i.e.*, subgraphs). However, their inter-subgraph similarities estimated with the outputs of local models are less reliable, because the final outputs of local models may not comprehensively represent the real distribution of subgraph data. In addition, they ignore the critical intra-heterogeneity which usually exists within each subgraph itself. To address these issues, we propose a novel **Fed**erated learning method by integrally modeling the **I**nter-**I**ntra **H**eterogeneity (FedIIH). For the inter-subgraph relationship, we propose a novel hierarchical variational model to infer the whole distribution of subgraph data in a multi-level form, so that we can accurately characterize the inter-subgraph similarities with the global perspective. For the intra-heterogeneity, we disentangle the subgraph into multiple latent factors and partition the model parameters into multiple parts, where each part corresponds to a single latent factor. Our FedIIH not only properly computes the distribution similarities between subgraphs, but also learns disentangled representations that are robust to irrelevant factors within subgraphs, so that it successfully considers the inter- and intra- heterogeneity simultaneously. Extensive experiments on six homophilic and five heterophilic graph datasets in both non-overlapping and overlapping settings demonstrate the effectiveness of our method when compared with nine state-of-the-art methods. Specifically, FedIIH averagely outperforms the second-best method by a large margin of 5.79% on all heterophilic datasets.

**Code** — https://github.com/blgpb/FedIIH

## Introduction

Graphs are ubiquitous data structures in lots of important domains such as social media, transportation, and recommendation systems (Tu et al. 2021, 2024a,b; Bai et al. 2022; Yu et al. 2023a,b; Zhou et al. 2024). In many real-world scenarios, a global graph is usually made up of multiple subgraphs that are distributed across devices, and subgraphs are only locally accessible due to privacy and regulatory concerns. Recently, Graph Federated Learning (GFL) has received increasing attention (Tan et al. 2023b; Huang et al. 2023), where each client individually trains a local model based on the corresponding subgraph, and a central server adaptively aggregates the models from all clients.

Since subgraphs are different parts of a global graph, there inevitably exists heterogeneity among them, making it difficult to realize federated collaboration. Ignoring this heterogeneity will degrade the performance of several traditional Federated Learning (FL) methods (*e.g.*, FedAvg (McMahan et al. 2017)) which rely heavily on the assumption that all clients have similar data distributions. To deal with this deficiency, recently, a number of personalized FL methods (Li et al. 2020; Pillutla et al. 2022; Tan et al. 2023a; Baek et al. 2023; Zhang et al. 2023) have been proposed. For example, in (Baek et al. 2023; Li et al. 2023; Zhang et al. 2024), personalized GFL methods estimate the pairwise similarities between clients, and then they perform weighted federations based on the client similarities.

However, the performance of most existing personalized GFL methods is still limited due to their improper similarity calculations and ignored intra-heterogeneity. First, most existing methods (Baek et al. 2023; Li et al. 2023; Zhang et al. 2024) compute the inter-subgraph similarities based on the simplex outputs of local models. Since the outputs of local models cannot accurately reveal the whole distribution of subgraph data, the calculated similarities are hardly generalizable, leading to negative impacts on the weighted federation. Moreover, although most existing methods successfully consider inter-heterogeneity, they ignore the crucial intra-heterogeneity (*i.e.*, the heterogeneity of intra-subgraph), which commonly exists in real-world graphs. Intra-heterogeneity can be defined as different types of connections in the subgraph on each client. For example, a user in a social graph is connected to others for various different reasons, such as families, hobbies, studies, and work. Meanwhile, as confirmed by (Xie et al. 2021), inter-heterogeneity is defined as divergent distributions of both graph structures and node features among different clients.

To address the above-mentioned issues, we propose a novel **Fed**erated learning method by integrally modeling the **I**nter-**I**ntra **H**eterogeneity (FedIIH), which is shown in the right panel of Fig. 1. On one hand, we propose a new Hi-

*Corresponding authors: Chen Gong and Shuo Chen.

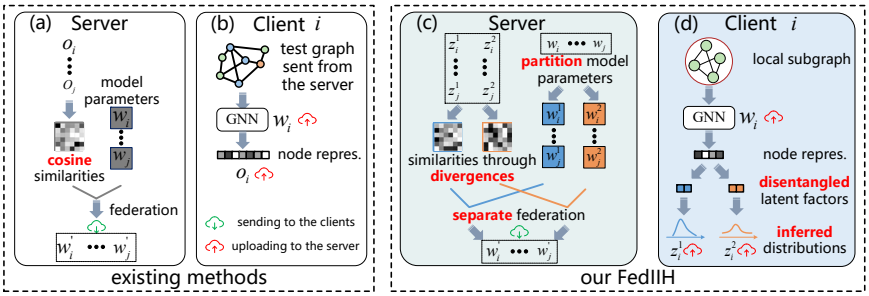



Figure 1: A framework comparison between existing methods and our FedIIH.

erarchical Variational Graph AutoEncoder (HVGAE) by using the hierarchical Bayesian model (Gelman and Hill 2006; Tran, Ranganath, and Blei 2017), so as to build the posterior dependencies between the latent factors of the local subgraphs and those of the global graph. Subsequently, we infer the subgraph data distribution and compute the similarities between pairwise clients (see Fig. 1c). On the other hand, we disentangle the local subgraph into multiple latent factors (see Fig. 1d). After that, the model parameters can be partitioned to correspond exactly to each latent factor, such that we can accordingly perform the separate federation based on our calculated similarities.

Our FedIIH not only properly computes the distribution similarities between subgraphs, but also learns disentangled representations robust to irrelevant factors within each subgraph, effectively alleviating the inter-intra heterogeneity and significantly improving the model performances on different types of graphs. Main contributions of our work are:

- For the inter-heterogeneity, we propose a novel method HVGAE to characterize the inter-heterogeneity from a multi-level global perspective, so that we can infer the data distributions of local subgraphs and properly compute the distribution similarities between subgraphs.
- For the intra-heterogeneity, we disentangle the subgraph into several latent factors, such that the federations can be separately performed, and this is the first time in GFL that considers the intra-heterogeneity.
- Extensive experiments on eleven datasets demonstrate the effectiveness of our proposed FedIIH, where our method averagely outperforms the second-best method by a large margin of 5.79% on all heterophilic graph data.

## Related Work

Here we first describe the preliminaries, and then review the typical work related to this paper, including GFL and personalized FL.

### Preliminaries

We focus on the task of node classification and aim to collaboratively train node classifiers with local subgraphs on distributed clients under the control of a server. For given $M$ clients, each of them has a local subgraph $\mathcal{G}_m = \langle \mathcal{V}_m, \mathcal{E}_m \rangle$, where $\mathcal{V}_m$ represents the node set, and $\mathcal{E}_m$ is the edge set $(m = 1, \ldots, M)$. The node feature matrix and adjacency matrix of $\mathcal{G}_m$ are denoted as $\mathbf{X}_m \in \mathbb{R}^{n_m \times d}$ and $\mathbf{A}_m \in \mathbb{R}^{n_m \times n_m}$, respectively. Here $n_m$ is the number of nodes in $\mathcal{G}_m$ and $d$ is the feature dimension. Due to the privacy constraints, $\mathcal{G}_m$ on each client is inaccessible to the others.

### Graph Federated Learning

FED-PUB (Baek et al. 2023) estimates similarities between subgraphs based on the outputs of local models. Then, it performs a weighted averaging of the local models for each client based on the similarities. Similarly, FedGTA (Li et al. 2023) and FedGT (Zhang et al. 2024) compute the similarities based on the mixed moments of processed neighbor features and embedding vectors, respectively. Then, they both perform the weighted federation, where model parameters with high similarities are assigned with larger weights during the weighted federation. However, the outputs of local models cannot faithfully reveal the whole distribution of subgraph data, such that the computed similarities are improper, which may impact the weighted federation, thus decreasing the performances. Consequently, we infer the whole distribution of subgraph data in a multi-level global perspective, so as to properly compute the similarities.

### Personalized Federated Learning

To deal with the heterogeneity, personalized FL methods (Li et al. 2020; Tan et al. 2023a; Arivazhagan et al. 2019) have obtained increasing attention. They can be categorized as similarity-based methods (Baek et al. 2023; Li et al. 2023; Zhang et al. 2024), local customization-based methods (Li et al. 2020; Arivazhagan et al. 2019; T. Dinh, Tran, and Nguyen 2020), and meta-learning-based methods (Chen et al. 2018; Fallah, Mokhtari, and Ozdaglar 2020). For similarity-based methods, they compute the inter-subgraph similarities and then perform weighted federation. In contrast, as one of the local customization-based methods, FedProx (Li et al. 2020) customizes a personalized model for each client by adding a proximal term as a subproblem. Similarly, FedPer (Arivazhagan et al. 2019) only federates the weights of the backbone while training the personalized classification layer in each client. However, these methods only consider the inter-heterogeneity, while ignoring the intra-heterogeneity. Therefore, in this paper, we propose to characterize both inter- and intra- heterogeneity.

## Methodology

This section details our proposed FedIIH. Specifically, we describe the modeling process of the intra-heterogeneity and

inter-heterogeneity, respectively.

## Modeling Intra-heterogeneity

Since there are diverse connecting relations among nodes in a real-world subgraph (Ma et al. 2019; Li et al. 2021; Guo et al. 2024), there is inevitably heterogeneity within the subgraph. To deal with the intra-heterogeneity, we aim to disentangle the subgraph into $K$ latent factors, which are utilized to represent different relations within the subgraph. In this paper, the well-known Disentangled Graph Convolutional Network (DisenGCN) (Ma et al. 2019) is considered for its simplicity. Here we use the node $u$ as an example to describe the disentangling process of DisenGCN. Given one node $i \in \{u\} \cup \{v|(u,v) \in \mathcal{G}_m\}$ in the subgraph $\mathcal{G}_m$, let $\mathbf{x}^i \in \mathbb{R}^d$ denote its node feature vector. This node feature $\mathbf{x}^i$ is first projected to $K$ subspaces, and its node representation in the $k$-th subspace can be represented by

$$\mathbf{z}^{i,k} = \frac{\sigma[(\mathbf{W}^k)^\top \mathbf{x}^i + \mathbf{b}^k]}{||\sigma[(\mathbf{W}^k)^\top \mathbf{x}^i + \mathbf{b}^k]||_2}, \tag{1}$$

where $\mathbf{W}^k \in \mathbb{R}^{d \times \frac{d_{\text{out}}}{K}}$ and $\mathbf{b}^k \in \mathbb{R}^{\frac{d_{\text{out}}}{K}}$ are learnable parameters. Here $\sigma[\cdot]$ denotes the nonlinear activation function, and $d_{\text{out}}$ denotes the output dimension of node representations. After the projection operation via Eq. (1), $\mathbf{z}^{i,k} \in \mathbb{R}^{\frac{d_{\text{out}}}{K}}$ represents the aspect of node $i$ that are related with the $k$-th latent factor. Second, DisenGCN proposes a neighborhood routing mechanism to identify the latent factor that causes the connection between node $u$ and its neighbor node $v$, and accordingly extract features of $v$ that are specific to that factor. With this mechanism, the node representation in each subspace is aggregated independently, such that we can obtain the disentangled latent factors $\{\mathbf{c}^{u,1}, \mathbf{c}^{u,2}, \cdots, \mathbf{c}^{u,K}\}$, where $\mathbf{c}^{u,k} \in \mathbb{R}^{\frac{d_{\text{out}}}{K}}$ represents the $k$-th aspect of node $u$. Note that there are no learnable parameters in the neighborhood routing mechanism. Finally, the disentangled node representation of $u$ can be obtained by

$$\mathbf{h}^u = \text{Con}(\mathbf{c}^{u,1}, \mathbf{c}^{u,2}, \cdots, \mathbf{c}^{u,K}), \tag{2}$$

where $\mathbf{h}^u \in \mathbb{R}^{d_{\text{out}}}$, and 'Con' denotes the concatenation operation performed along the column. Similarly, we can obtain all of the disentangled node representations in $\mathcal{G}_m$ based on Eq. (2). After that, we use the matrix $\mathbf{H}_m \in \mathbb{R}^{n_m \times d_{\text{out}}}$ to denote the disentangled node representations in $\mathcal{G}_m$. According to Eq. (1), $\mathbf{W}^k$ and $\mathbf{b}^k$ correspond exactly to the $k$-th latent factor. Consequently, we perform the separate federation with parameters (*i.e.*, $\mathbf{W}^k$ and $\mathbf{b}^k$) specific to the $k$-th latent factor, which is described in the following section.

## Modeling Inter-heterogeneity

Since the local views of subgraphs may not be sufficient to estimate the inter-heterogeneity, we aim to model the inter-heterogeneity from a multi-level global perspective. Moreover, since the similarities between clients are essentially determined by the similarities of local subgraph data distributions, we seek to infer the data distribution of subgraphs and thereby compute the similarities between clients through divergences. Specifically, we propose the HVGAE to infer the subgraph data distribution of the $m$-th client based on

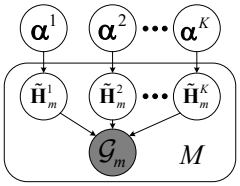



Figure 2: The graphical model of our proposed HVGAE, where $\tilde{\mathbf{H}}_m^1, \tilde{\mathbf{H}}_m^2, \cdots, \tilde{\mathbf{H}}_m^K$ denote the local latent factors on the $m$-th client, and $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \cdots, \boldsymbol{\alpha}^K$ denote global latent factors on the server.

the disentangled node representations (*i.e.*, $\mathbf{H}_m$). For ease of expression, we rewrite $\mathbf{H}_m$ as $\mathbf{H}_m^1, \mathbf{H}_m^2, \cdots, \mathbf{H}_m^K$, where $\mathbf{H}_m^k \in \mathbb{R}^{n_m \times \frac{d_{\text{out}}}{K}}$ denotes the disentangled node representations with the $k$-th latent factor. As $\mathbf{H}_m^1, \mathbf{H}_m^2, \cdots, \mathbf{H}_m^K$ are deterministic results output by DisenGCN, we use $\tilde{\mathbf{H}}_m^1, \tilde{\mathbf{H}}_m^2, \cdots, \tilde{\mathbf{H}}_m^K$ to represent random latent variables.

**Hierarchical model** To build the posterior dependencies between the latent factors of the local subgraphs and those of the global graph, we assume that the latent factors of the subgraphs are governed by those of the global graph based on the theory of hierarchical Bayesian model (Gelman and Hill 2006; Tran, Ranganath, and Blei 2017). As shown in Fig. 2, $\tilde{\mathbf{H}}_m^1, \tilde{\mathbf{H}}_m^2, \cdots, \tilde{\mathbf{H}}_m^K$ on the $m$-th client are local latent factors, which are linked via the higher-level global latent factors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \cdots, \boldsymbol{\alpha}^K$ on the server, respectively. Note that $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \cdots, \boldsymbol{\alpha}^K$ are shared across all clients.

Based on the graphical model in Fig. 2, the joint probability can be formulated as

$$p(\mathcal{G}_{1:M}, \tilde{\mathbf{H}}_{1:M}^1, \tilde{\mathbf{H}}_{1:M}^2, \cdots, \tilde{\mathbf{H}}_{1:M}^K, \boldsymbol{\alpha}^{1:K}) = \prod_{k=1}^{K} p(\boldsymbol{\alpha}^k) \prod_{m=1}^{M} \prod_{k'=1}^{K} p(\mathcal{G}_m|\tilde{\mathbf{H}}_m^1, \tilde{\mathbf{H}}_m^2, \cdots, \tilde{\mathbf{H}}_m^K) p(\tilde{\mathbf{H}}_m^k|\boldsymbol{\alpha}^{k'}), \tag{3}$$

where $\mathcal{G}_{1:M}$ is the abbreviation for $(\mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_M)$. $(\tilde{\mathbf{H}}_{1:M}^1, \tilde{\mathbf{H}}_{1:M}^2, \cdots, \tilde{\mathbf{H}}_{1:M}^K)$ is the abbreviation for $(\tilde{\mathbf{H}}_1^1, \tilde{\mathbf{H}}_2^1, \cdots, \tilde{\mathbf{H}}_M^1, \tilde{\mathbf{H}}_1^2, \tilde{\mathbf{H}}_2^2, \cdots, \tilde{\mathbf{H}}_M^2, \tilde{\mathbf{H}}_1^K, \tilde{\mathbf{H}}_2^K, \cdots, \tilde{\mathbf{H}}_M^K)$, $\boldsymbol{\alpha}^{1:K}$ is the abbreviation for $(\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \cdots, \boldsymbol{\alpha}^K)$, and $p(\boldsymbol{\alpha}^k)$ denotes the prior distribution of $\boldsymbol{\alpha}^k$. In Eq. (3), $p(\mathcal{G}_m|\tilde{\mathbf{H}}_m^1, \tilde{\mathbf{H}}_m^2, \cdots, \tilde{\mathbf{H}}_m^K)$ denotes the conditional distribution, and $p(\tilde{\mathbf{H}}_m^k|\boldsymbol{\alpha}^{k'})$ denotes the prior distribution of $\tilde{\mathbf{H}}_m^k$. Moreover, the true posterior distribution of both local and global latent factors can be formulated as $p(\tilde{\mathbf{H}}_{1:M}^1, \tilde{\mathbf{H}}_{1:M}^2, \cdots, \tilde{\mathbf{H}}_{1:M}^K, \boldsymbol{\alpha}^{1:K}|\mathcal{G}_{1:M})$. However, this true posterior distribution is computationally intractable. Therefore, we attempt to approximate it with a tractable approximate posterior distribution $q(\tilde{\mathbf{H}}_{1:M}^1, \tilde{\mathbf{H}}_{1:M}^2, \cdots, \tilde{\mathbf{H}}_{1:M}^K, \boldsymbol{\alpha}^{1:K}|\mathcal{G}_{1:M})$, which can be formulated as

$$\begin{aligned} &q(\tilde{\mathbf{H}}_{1:M}^1, \tilde{\mathbf{H}}_{1:M}^2, \cdots, \tilde{\mathbf{H}}_{1:M}^K, \boldsymbol{\alpha}^{1:K}|\mathcal{G}_{1:M}) \\ &= \prod_{k=1}^{K} q(\boldsymbol{\alpha}^k) \cdot \prod_{m=1}^{M} \prod_{k'=1}^{K} q(\tilde{\mathbf{H}}_m^{k'}|\mathcal{G}_m), \end{aligned} \tag{4}$$

where $q(\boldsymbol{\alpha}^k)$ denotes the marginal distribution, and $q(\tilde{\mathbf{H}}_m^{k'}|\mathcal{G}_m)$ is the approximate posterior distribution of $\tilde{\mathbf{H}}_m^{k'}$.

**Variational inference** Since the approximate posterior distributions in Eq. (4) are still intractable, we propose to use the variational inference (Kingma and Welling 2013; Zhang et al. 2019) to infer them. According to the graphical model in Fig. 2, the Evidence Lower BOund (ELBO) can be derived as follows:

$$\mathcal{L}_{\mathrm{ELBO}} = \sum_{k=1}^{K} \log p(\tilde{\boldsymbol{\alpha}}^k) + \sum_{m=1}^{M} \Big\{ \mathbb{E}_{q(\tilde{\mathbf{H}}_m|\mathcal{G}_m)} \big[ \log p(\mathcal{G}_m|\tilde{\mathbf{H}}_m) \big] - \sum_{k'=1}^{K} D_{\mathrm{KL}}\big(q(\tilde{\mathbf{H}}_m^{k'}|\mathcal{G}_m)||p(\tilde{\mathbf{H}}_m^{k'}|\tilde{\boldsymbol{\alpha}}^{k'})\big) \Big\}, \tag{5}$$

where $\log p(\tilde{\boldsymbol{\alpha}}^k)$ denotes the log prior probability of $\tilde{\boldsymbol{\alpha}}^k$. Here $\tilde{\boldsymbol{\alpha}}^k$ denotes the posterior mean of $\boldsymbol{\alpha}^k$ for the $k$-th global latent factor, and $D_{\mathrm{KL}}$ denotes the Kullback-Leibler (KL) divergence (Moreno, Ho, and Vasconcelos 2003).

**Adaptation to FL** However, since the server does not have access to the private data of the clients, the optimization process on the server (*i.e.*, $\sum_{k=1}^{K} \log p(\tilde{\boldsymbol{\alpha}}^k)$) is challenging. To adapt Eq. (5) to the federated learning scenario, we make some modifications to $\sum_{k=1}^{K} \log p(\tilde{\boldsymbol{\alpha}}^k)$. On one hand, we substitute $\sum_{k=1}^{K} \log p(\tilde{\boldsymbol{\alpha}}^k)$ by $\sum_{m=1}^{M} \sum_{k=1}^{K} \log p(\hat{\boldsymbol{\alpha}}_m^k)$, where $\hat{\boldsymbol{\alpha}}_m^k$ denotes the learnable parameter deployed on the $m$-th client. On the other hand, $\hat{\boldsymbol{\alpha}}_m^k$ on each client is constrained by the KL divergence between $p(\hat{\boldsymbol{\alpha}}_m^k)$ and $p(\tilde{\boldsymbol{\alpha}}^k)$. Finally, we have that

$$\hat{\mathcal{L}}_{\mathrm{ELBO}} = \sum_{m=1}^{M} \sum_{k=1}^{K} \Big\{ \log p(\hat{\boldsymbol{\alpha}}_m^k) - D_{\mathrm{KL}}\big(p(\hat{\boldsymbol{\alpha}}_m^k)||p(\tilde{\boldsymbol{\alpha}}^k)\big) \Big\} + \sum_{m'=1}^{M} \Big\{ \mathbb{E}_{q(\tilde{\mathbf{H}}_{m'}|\mathcal{G}_{m'})} \big[ \log p(\mathcal{G}_{m'}|\tilde{\mathbf{H}}_{m'}) \big] - \sum_{k'=1}^{K} D_{\mathrm{KL}}\big(q(\tilde{\mathbf{H}}_{m'}^{k'}|\mathcal{G}_{m'})||p(\tilde{\mathbf{H}}_{m'}^{k'}|\tilde{\boldsymbol{\alpha}}^{k'})\big) \Big\}. \tag{6}$$

Eq. (6) implies that the optimizations of ELBO can be distributedly performed on $M$ clients in a parallel way.

**Variational graph autoencoder** To implement the approximate posterior distribution $q(\tilde{\mathbf{H}}_m^k|\mathcal{G}_m)$ and the mathematical expectation $\mathbb{E}_{q(\tilde{\mathbf{H}}_m|\mathcal{G}_m)} \big[ \log p(\mathcal{G}_m|\tilde{\mathbf{H}}_m) \big]$ in Eq. (6), we introduce a simple yet effective framework (*i.e.*, Variational Graph AutoEncoder (VGAE) (Kipf and Welling 2016)) on each client, which includes an inference network (*a.k.a.* encoder) and a generative network (*a.k.a.* decoder). First, we take an inference model parameterized by two DisenGCNs (*i.e.*, $\mathrm{DisenGCN}_{\boldsymbol{\mu}_m}(\mathcal{G}_m)$ and $\mathrm{DisenGCN}_{\boldsymbol{\sigma}_m}(\mathcal{G}_m)$). Note that $\mathrm{DisenGCN}_{\boldsymbol{\mu}_m}(\mathcal{G}_m)$ and $\mathrm{DisenGCN}_{\boldsymbol{\sigma}_m}(\mathcal{G}_m)$ are used to infer the means and standard deviations of $\mathcal{G}_m$ for $K$ latent factors (*i.e.*, $\mathbf{H}_{m,\boldsymbol{\mu}}$ and $\mathbf{H}_{m,\boldsymbol{\sigma}}$), respectively. For ease of expression, we rewrite them as $\mathbf{H}_{m,\boldsymbol{\mu}}^1, \mathbf{H}_{m,\boldsymbol{\mu}}^2, \cdots, \mathbf{H}_{m,\boldsymbol{\mu}}^K$ and $\mathbf{H}_{m,\boldsymbol{\sigma}}^1, \mathbf{H}_{m,\boldsymbol{\sigma}}^2, \cdots, \mathbf{H}_{m,\boldsymbol{\sigma}}^K$, respectively. By using the reparameterization trick (Kingma and Welling 2013), we can have

$$\tilde{\mathbf{H}}_m^k = \mathbf{H}_{m,\boldsymbol{\mu}}^k + \mathbf{H}_{m,\boldsymbol{\sigma}}^k \odot \boldsymbol{\epsilon}, \tag{7}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and $\odot$ denotes the element-wise product. Second, our generative network is constructed by an inner product between latent variables. Since $\mathbb{E}_{q(\tilde{\mathbf{H}}_m|\mathcal{G}_m)} \big[ \log p(\mathcal{G}_m|\tilde{\mathbf{H}}_m) \big]$ in Eq. (6) can be regarded as a reconstruction loss, we implement it by

$$\mathbb{E}_{q(\tilde{\mathbf{H}}_m|\mathcal{G}_m)} \big[ \log p(\mathcal{G}_m|\tilde{\mathbf{H}}_m) \big] = p(\mathbf{A}_m|\tilde{\mathbf{H}}_m) = \prod_{i=1}^{n_m} \prod_{j=1}^{n_m} p(\mathbf{A}_m^{ij}|\mathbf{r}_i, \mathbf{r}_j), \tag{8}$$

where $p(\mathbf{A}_m^{ij} = 1|\mathbf{r}_i, \mathbf{r}_j) = \sigma(\mathbf{r}_i^\top \mathbf{r}_j)$, and $\mathbf{A}_m^{ij}$ denotes the element of $\mathbf{A}_m$. In Eq. (8), $\mathbf{r}_i$ and $\mathbf{r}_j$ are the $i$-th and $j$-th rows of the matrix $\tilde{\mathbf{H}}_m$, respectively.

**Similarity calculation** Since the similarities between clients are essentially determined by the similarities of the local data distributions, we seek to calculate the similarities between clients based on the divergences of the inferred subgraph data distributions. Moreover, since HVGAE has $K$ disentangled latent factors, we can compute the similarities corresponding to each latent factor separately. Specifically, the similarity with respect to the $k$-th latent factor between clients $m$ and $j$ can be computed as

$$S(m,j)^k = 1 - D_{\mathrm{JS}}\big(q(\tilde{\mathbf{H}}_m^k|\mathcal{G}_m)||q(\tilde{\mathbf{H}}_j^k|\mathcal{G}_j)\big), \tag{9}$$

where $S(m,j)^k$ denotes the computed similarity, and $D_{\mathrm{JS}}$ represents the Jensen-Shannon (JS) divergence (Sutter, Daunhawer, and Vogt 2020).

## Federated Aggregation

Before describing the federation process, let us analyze the learnable parameters that should be federated in our FedIIH. The first part of the federated parameters are $\mathbf{W}_m^1, \mathbf{W}_m^2, \cdots, \mathbf{W}_m^K$ and $\mathbf{b}_m^1, \mathbf{b}_m^2, \cdots, \mathbf{b}_m^K$ in Eq. (1). The second part of the federated parameters are $\mathbf{W}_m^{\mathrm{cls}} \in \mathbb{R}^{c \times d_{\mathrm{out}}}$ and $\mathbf{b}_m^{\mathrm{cls}} \in \mathbb{R}^c$, which come from the node classifier. Here $c$ is the number of node classes. For the federated aggregation of $\mathbf{W}_m^1, \mathbf{W}_m^2, \cdots, \mathbf{W}_m^K$, $\mathbf{b}_m^1, \mathbf{b}_m^2, \cdots, \mathbf{b}_m^K$, and $\mathbf{W}_m^{\mathrm{cls}}$, we propose a separate federation method. Specifically, with the computed similarity $S(i,j)^k$, we use the weighted averaging of parameters across different clients. Moreover, since HVGAE has $K$ disentangled latent factors, and parameters (*e.g.*, $\mathbf{W}^k$ and $\mathbf{b}^k$ in Eq. (1)) correspond exactly to the $k$-th latent factor, we can therefore perform the separate federation according to each latent factor. For ease of expression, we rewrite $\mathbf{W}_m^{\mathrm{cls}}$ as $\mathbf{W}_m^{\mathrm{cls},1}, \mathbf{W}_m^{\mathrm{cls},2}, \cdots, \mathbf{W}_m^{\mathrm{cls},K}$, where $\mathbf{W}_m^{\mathrm{cls},k} \in \mathbb{R}^{c \times \frac{d_{\mathrm{out}}}{K}}$ denotes the parameters with respect to the $k$-th latent factor. Our proposed separate federation can be defined as

$$\overline{\mathbf{W}}_m^k \leftarrow \sum_{j=1}^{M} \beta_{mj}^k \cdot \mathbf{W}_j^k, \quad \overline{\mathbf{b}}_m^k \leftarrow \sum_{j=1}^{M} \beta_{mj}^k \cdot \mathbf{b}_j^k, \quad \overline{\mathbf{W}}_m^{\mathrm{cls},k} \leftarrow \sum_{j=1}^{M} \beta_{mj}^k \cdot \mathbf{W}_j^{\mathrm{cls},k}, \tag{10}$$

where

$$\beta_{mj}^{k} = \frac{\exp\left(\tau \cdot S(m,j)^{k}\right)}{\sum_{l=1}^{M} \exp\left(\tau \cdot S(m,l)^{k}\right)}, \tag{11}$$

and $\beta_{mj}^{k}$ denotes the normalized similarity with respect to the $k$-th latent factor between clients $m$ and $j$. In Eq. (11), $\tau$ denotes a hyperparameter for scaling the similarity score. Our proposed separate federation not only allows different clients to obtain personalized parameters, which is beneficial for dealing with inter-heterogeneity, but also allows parameters in the same client to be federated separately according to each latent factor, which is beneficial for dealing with intra-heterogeneity. For the federation of the remaining parameter $\mathbf{b}_{m}^{\text{cls}} \in \mathbb{R}^{c}$, we use the federation method as proposed in FedAvg (McMahan et al. 2017), which is simple but effective.

## Experiments

To validate the effectiveness of our proposed FedIIH, we perform extensive experiments on eleven widely used benchmark datasets, including both homophilic and heterophilic graphs. We use both the non-overlapping and overlapping subgraph partitioning settings.

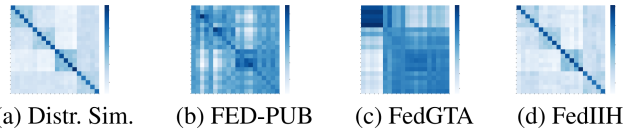
### Main Results

**Homophilic datasets** Tab. 1 and Tab. 2 show the node classification results on the homophilic datasets in two partitioning settings, respectively. Our FedIIH achieves the best performance among all the methods, and the standard deviations are also relatively small as well, suggesting that FedIIH is more effective and stable than the compared methods. Moreover, the average accuracy of FedIIH for all six datasets in the non-overlapping scenario is 83.10%, which is 1.51% higher than the second-best method. In the overlapping scenario, the average accuracy of FedIIH is 81.01%, which is 1.48% higher than the second-best method.

**Heterophilic datasets** Tab. 3 and Tab. 4 show the node classification results on the heterophilic datasets in two partitioning settings, respectively. FedIIH not only achieves the best average performance among all baseline methods, but also outperforms the second-best method (*i.e.*, FedSage+) by 5.79% and 4.53% in the non-overlapping and overlapping scenarios, respectively. This is because our FedIIH not only considers the inter-heterogeneity as these methods do, but also successfully deals with the intra-heterogeneity that they ignore. Moreover, although the intra-heterogeneity on the heterophilic datasets is stronger than that on the homophilic datasets (Platonov et al. 2023), FedIIH achieves a larger performance improvement than on the homophilic datasets.

### Ablation Study

Our FedIIH employs the hierarchical model and variational inference to compute the inter-subgraph similarities. In addition, we use the disentanglement to learn disentangled representations. To shed light on the contributions of these components, we report the experimental results of FedIIH when each of these components is removed on the *Cora* and *Amazon-ratings* datasets in Tab. 5. For simplicity, 'w/o HM', 'w/o VI', and 'w/o Dis' denote the reduced models by removing the hierarchical model, variational inference, and disentanglement, respectively. The performance decreases when any component is removed, showing that each component contributes a lot to the final performance.



(a) Distr. Sim. (b) FED-PUB (c) FedGTA (d) FedIIH

Figure 3: Similarity heatmaps on the *Amazon-ratings* dataset in the overlapping setting with 20 clients.

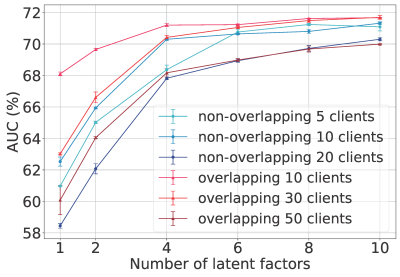



Figure 4: Since the intra-heterogeneity of the *Tolokers* dataset is high, the performances increase consistently as the value of $K$ increases.

### Effectiveness of Inter-Subgraph Similarity and $K$

First, we validate the effectiveness of the inter-subgraph similarities calculated by our method. In Fig. 3, we show the similarity heatmaps on the *Amazon-ratings* dataset in the overlapping setting with 20 clients. We compute the subgraph distribution similarity by using both semantic and structural information, and we treat such distribution similarity (Fig. 3a) as the ground truth. Fig. 3d is close to the ground truth. In contrast, the similarity heatmaps of FED-PUB (Fig. 3b) and FedGTA (Fig. 3c) are both different from the ground truth, which implies that their calculated similarities are not good enough. Second, we investigate the effectiveness of $K$. As shown in Fig. 4, the performances on the *Tolokers* dataset increase consistently as the value of $K$ increases. It means that disentangling the subgraph into several latent factors is beneficial to improve the performances.

## Conclusion

In this paper, we proposed a novel method FedIIH, which naturally integrates the inter- and intra- heterogeneity in GFL. On one hand, our new method characterizes the inter-heterogeneity from a multi-level global perspective, and thus it can properly compute the inter-subgraph similarities based on the whole distribution. On the other hand, it disentangles the subgraph into several latent factors, so that we can further consider the critical intra-heterogeneity. To the best of our knowledge, this is the first time in GFL that combines both inter- and intra- heterogeneity into a unified framework. Due to the adequate consideration of inter-intra heterogeneity, our method achieves satisfactory results on eleven datasets and outperforms the second-best method by a large margin of 5.79% on the heterophilic graph data.

| | Cora | | | CiteSeer | | | PubMed | | | - |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | - |
| Local | 81.30±0.21 | 79.94±0.24 | 80.30±0.25 | 69.02±0.05 | 67.82±0.13 | 65.98±0.17 | 84.04±0.18 | 82.81±0.39 | 82.65±0.03 | - |
| FedAvg | 74.45±5.64 | 69.19±0.67 | 69.50±3.58 | 71.06±0.60 | 63.61±3.59 | 64.68±1.83 | 79.40±0.11 | 82.71±0.29 | 80.97±0.26 | - |
| FedProx | 72.03±4.56 | 60.18±7.04 | 48.22±6.18 | 71.73±1.11 | 63.33±3.25 | 64.85±1.35 | 79.45±0.25 | 82.55±0.24 | 80.50±0.25 | - |
| FedPer | 81.68±0.40 | 79.35±0.04 | 78.01±0.32 | 70.41±0.32 | 70.53±0.28 | 66.64±0.27 | 85.80±0.21 | 84.20±0.28 | 84.72±0.31 | - |
| GCFL | 81.47±0.65 | 78.66±0.27 | 79.21±0.70 | 70.34±0.57 | 69.01±0.12 | 66.33±0.05 | 85.14±0.33 | 84.18±0.19 | 83.94±0.36 | - |
| FedGNN | 81.51±0.68 | 70.12±0.99 | 70.10±3.52 | 69.06±0.92 | 55.52±3.17 | 52.23±6.00 | 79.52±0.23 | 83.25±0.45 | 81.61±0.59 | - |
| FedSage+ | 72.97±5.94 | 69.05±1.59 | 57.97±12.6 | 70.74±0.69 | 65.63±3.10 | 65.46±0.74 | 79.57±0.24 | 82.62±0.31 | 80.82±0.25 | - |
| FED-PUB | 83.70±0.19 | 81.54±0.12 | 81.75±0.56 | 72.68±0.44 | 72.35±0.53 | 67.62±0.12 | 86.79±0.09 | 86.28±0.18 | 85.53±0.30 | - |
| FedGTA | 80.06±0.63 | 80.59±0.38 | 79.01±0.31 | 70.12±0.10 | 71.57±0.34 | 69.94±0.14 | 87.75±0.01 | 86.80±0.01 | 87.12±0.05 | - |
| AdaFGL | 82.01±0.51 | 80.09±0.00 | 79.74±0.05 | 71.44±0.27 | 72.34±0.00 | 70.95±0.45 | 86.91±0.28 | 86.97±0.10 | 86.59±0.21 | - |
| FedIIH (Ours) | **84.11±0.17** | **81.85±0.09** | **83.01±0.15** | **72.86±0.25** | **76.50±0.06** | **73.36±0.41** | **87.80±0.18** | **87.65±0.18** | **87.19±0.25** | - |
| | Amazon-Computer | | | Amazon-Photo | | | ogbn-arxiv | | | Avg. |
| Methods | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | All |
| Local | 89.22±0.13 | 88.91±0.17 | 89.52±0.20 | 91.67±0.09 | 91.80±0.02 | 90.47±0.15 | 66.76±0.07 | 64.92±0.09 | 65.06±0.05 | 79.57 |
| FedAvg | 84.88±1.96 | 79.54±0.23 | 74.79±0.24 | 89.89±0.83 | 83.15±3.71 | 81.35±1.04 | 65.54±0.07 | 64.44±0.10 | 63.24±0.13 | 74.58 |
| FedProx | 85.25±1.27 | 83.81±1.09 | 73.05±1.30 | 90.38±0.48 | 80.92±4.64 | 82.32±0.29 | 65.21±0.20 | 64.37±0.18 | 63.03±0.04 | 72.84 |
| FedPer | 89.67±0.34 | 89.73±0.04 | 87.86±0.43 | 91.44±0.37 | 91.76±0.23 | 90.59±0.06 | 66.87±0.05 | 64.99±0.18 | 64.66±0.11 | 79.94 |
| GCFL | 89.07±0.91 | 90.03±0.16 | 89.08±0.25 | 91.99±0.29 | 92.06±0.25 | 90.79±0.17 | 66.80±0.12 | 65.09±0.08 | 65.08±0.04 | 79.90 |
| FedGNN | 88.08±0.15 | 88.18±0.41 | 83.16±0.13 | 90.25±0.70 | 87.12±2.01 | 81.00±4.48 | 65.47±0.22 | 64.21±0.32 | 63.80±0.05 | 75.23 |
| FedSage | 85.04±0.61 | 80.50±1.13 | 70.42±0.85 | 90.77±0.44 | 76.81±8.24 | 80.58±1.15 | 65.69±0.09 | 64.52±0.14 | 63.31±0.20 | 73.47 |
| FED-PUB | **90.74±0.05** | 90.55±0.13 | 90.12±0.09 | 93.29±0.19 | 92.73±0.18 | 91.92±0.12 | 67.77±0.09 | 66.58±0.08 | 66.64±0.12 | 81.59 |
| FedGTA | 86.69±0.18 | 86.66±0.23 | 85.01±0.87 | 93.33±0.12 | 93.50±0.21 | 92.61±0.15 | 60.32±0.04 | 60.22±0.09 | 58.74±0.14 | 79.45 |
| AdaFGL | 80.20±0.05 | 83.62±0.26 | 84.53±0.23 | 86.69±0.19 | 89.85±0.83 | 88.11±0.05 | 52.73±0.19 | 51.77±0.36 | 50.94±0.08 | 76.97 |
| FedIIH (Ours) | **90.74±0.13** | **90.86±0.23** | **90.44±0.05** | **93.42±0.02** | **94.22±0.08** | **93.55±0.09** | **70.30±0.06** | **69.34±0.02** | **68.65±0.04** | **83.10** |

Table 1: Node classification results of different methods on the **homophilic** graph datasets in the **non-overlapping** subgraph partitioning setting. The best results are shown in **bold**.

| | Cora | | | CiteSeer | | | PubMed | | | - |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | - |
| Local | 73.98±0.25 | 71.65±0.12 | 76.63±0.10 | 65.12±0.08 | 64.54±0.42 | 66.68±0.44 | 82.32±0.07 | 80.72±0.16 | 80.54±0.11 | - |
| FedAvg | 76.48±0.36 | 53.99±0.98 | 53.99±4.53 | 69.48±0.15 | 66.15±0.64 | 66.51±1.00 | 82.67±0.11 | 82.05±0.12 | 80.24±0.35 | - |
| FedProx | 77.85±0.50 | 51.38±1.74 | 56.27±9.04 | 69.39±0.35 | 66.11±0.75 | 66.53±0.43 | 82.63±0.17 | 82.13±0.13 | 80.50±0.46 | - |
| FedPer | 78.73±0.31 | 74.18±0.24 | 74.42±0.37 | 69.81±0.28 | 65.19±0.81 | 67.64±0.44 | 85.31±0.06 | 84.35±0.38 | 83.94±0.10 | - |
| GCFL | 78.84±0.26 | 73.41±0.27 | 76.63±0.16 | 69.48±0.39 | 64.92±0.18 | 65.98±0.30 | 83.59±0.25 | 80.77±0.12 | 81.36±0.11 | - |
| FedGNN | 70.63±0.83 | 61.38±2.33 | 56.91±0.82 | 68.72±0.39 | 59.98±1.52 | 58.98±0.98 | 84.25±0.07 | 82.02±0.22 | 81.85±0.10 | - |
| FedSage | 77.52±0.46 | 51.99±0.42 | 55.48±11.5 | 68.75±0.48 | 65.97±0.02 | 65.93±0.30 | 82.77±0.08 | 82.14±0.11 | 80.31±0.68 | - |
| FED-PUB | 79.60±0.12 | 75.40±0.54 | 77.84±0.23 | 70.58±0.20 | 68.33±0.45 | 69.21±0.30 | 85.70±0.08 | 85.16±0.10 | 84.84±0.12 | - |
| FedGTA | 76.42±0.62 | 75.63±0.33 | 77.69±0.14 | 70.43±0.08 | 71.71±0.33 | 69.19±0.32 | 85.34±0.42 | 84.99±0.05 | 84.47±0.06 | - |
| AdaFGL | 78.50±0.19 | 75.80±0.23 | 74.41±0.00 | 72.63±0.15 | 68.18±0.31 | 62.90±0.75 | 85.58±0.23 | 85.85±0.41 | 84.45±0.07 | - |
| FedIIH (Ours) | **80.57±0.23** | **76.82±0.24** | **78.58±0.25** | **73.16±0.18** | **72.27±0.21** | **69.56±0.11** | **85.87±0.03** | **86.65±0.11** | **85.65±0.12** | - |
| | Amazon-Computer | | | Amazon-Photo | | | ogbn-arxiv | | | Avg. |
| Methods | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | All |
| Local | 88.50±0.20 | 86.66±0.00 | 87.04±0.02 | 92.17±0.12 | 90.16±0.12 | 90.42±0.15 | 62.52±0.07 | 61.32±0.04 | 60.04±0.04 | 76.72 |
| FedAvg | 88.99±0.19 | 83.37±0.47 | 76.34±0.12 | 92.91±0.07 | 89.30±0.22 | 74.19±0.57 | 63.56±0.02 | 59.72±0.06 | 60.94±0.24 | 73.38 |
| FedProx | 88.84±0.20 | 83.84±0.89 | 76.60±0.47 | 92.67±0.19 | 89.17±0.40 | 72.36±2.06 | 63.52±0.11 | 59.86±0.16 | 61.12±0.04 | 73.38 |
| FedPer | 89.30±0.04 | 87.99±0.23 | 88.22±0.27 | 92.88±0.24 | 91.23±0.16 | 90.92±0.38 | 63.97±0.08 | 62.29±0.04 | 61.24±0.11 | 78.42 |
| GCFL | 89.01±0.22 | 87.24±0.09 | 87.02±0.22 | 92.45±0.10 | 90.58±0.11 | 90.54±0.08 | 63.24±0.02 | 61.66±0.10 | 60.32±0.01 | 77.61 |
| FedGNN | 88.15±0.09 | 87.00±0.10 | 83.96±0.88 | 91.47±0.11 | 87.91±1.34 | 78.90±6.46 | 63.08±0.19 | 60.09±0.04 | 60.51±0.11 | 73.66 |
| FedSage | 89.24±0.15 | 81.33±1.20 | 76.72±0.39 | 92.76±0.05 | 88.69±0.99 | 72.41±1.36 | 63.24±0.02 | 59.90±0.12 | 60.95±0.09 | 73.12 |
| FED-PUB | 89.98±0.08 | 89.15±0.06 | 88.76±0.14 | 93.22±0.07 | 92.01±0.07 | 91.71±0.11 | 64.18±0.04 | 63.34±0.12 | 62.55±0.12 | 79.53 |
| FedGTA | 90.10±0.18 | 88.79±0.27 | 88.15±0.21 | 93.13±0.14 | 92.49±0.06 | 91.77±0.06 | 55.98±0.09 | 56.76±0.07 | 57.89±0.09 | 74.40 |
| AdaFGL | 80.49±0.00 | 80.42±0.00 | 82.12±0.00 | 89.24±0.00 | 88.34±0.00 | 87.68±0.00 | 56.81±0.06 | 55.17±0.00 | 54.82±0.00 | 75.74 |
| FedIIH (Ours) | **90.15±0.04** | **89.56±0.19** | **89.99±0.00** | **93.38±0.00** | **94.17±0.04** | **93.25±0.16** | **66.69±0.09** | **66.10±0.03** | **65.67±0.06** | **81.01** |

Table 2: Node classification results of different methods on the **homophilic** graph datasets in the **overlapping** subgraph partitioning setting. The best results are shown in **bold**.

| | Roman-empire | | | Amazon-ratings | | | Minesweeper | | | - |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | - |
| Local | 33.65±0.13 | 28.42±0.26 | 23.89±0.32 | **45.03±0.31** | **45.89±0.19** | **46.02±0.02** | 71.35±0.17 | 69.96±0.16 | 69.31±0.09 | - |
| FedAvg | 38.93±0.32 | 35.43±0.32 | 32.00±0.39 | 41.26±0.53 | 41.66±0.14 | 42.20±0.21 | 72.60±0.08 | 71.84±0.02 | 71.36±0.16 | - |
| FedProx | 27.95±0.59 | 26.43±1.41 | 23.12±0.49 | 36.92±0.00 | 36.86±0.14 | 36.96±0.00 | 71.91±0.27 | 70.66±0.20 | 71.50±0.37 | - |
| FedPer | 20.75±1.75 | 15.51±1.13 | 15.45±2.76 | 36.62±0.30 | 32.34±1.01 | 36.96±0.00 | 58.73±10.45 | 65.35±7.02 | 53.80±11.40 | - |
| GCFL | 30.40±0.16 | 29.44±0.49 | 26.73±0.19 | 36.92±0.00 | 36.86±0.14 | 36.96±0.00 | 72.04±0.13 | 71.14±0.09 | 47.77±0.14 | - |
| FedGNN | 30.26±0.11 | 29.09±0.01 | 26.60±0.02 | 36.92±0.00 | 36.72±0.00 | 36.96±0.00 | 72.03±0.13 | 71.12±0.09 | 71.71±0.27 | - |
| FedSage+ | 57.26±0.00 | 49.07±0.00 | 38.36±0.00 | 36.82±0.00 | 36.71±0.00 | 37.03±0.00 | **77.74±0.00** | 72.80±0.00 | 69.70±0.00 | - |
| FED-PUB | 40.80±0.26 | 36.77±0.30 | 32.67±0.39 | 44.41±0.41 | 44.85±0.17 | 45.39±0.50 | 72.18±0.02 | 71.56±0.05 | 70.72±0.40 | - |
| FedGTA | 61.56±0.27 | 60.94±0.19 | 59.65±0.28 | 41.22±0.66 | 39.40±0.44 | 39.24±0.12 | 45.60±1.41 | 64.97±0.35 | 49.63±8.64 | - |
| AdaFGL | 67.64±0.18 | 64.55±0.00 | 62.42±0.26 | 41.70±0.06 | 42.30±0.00 | 42.59±0.14 | 47.45±2.10 | 65.59±0.56 | 51.48±7.14 | - |
| FedIIH (Ours) | **68.32±0.05** | **66.44±0.28** | **64.61±0.13** | 44.26±0.24 | 44.24±0.10 | 45.19±0.04 | 74.29±0.02 | **73.23±0.04** | **72.81±0.02** | |
| | Tolokers | | | Questions | | | Avg. | | | |
| Methods | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | 5 Clients | 10 Clients | 20 Clients | All |
| Local | 67.81±0.17 | 70.04±0.23 | 62.34±0.67 | 66.73±0.57 | 57.96±0.10 | 60.00±0.21 | 56.91 | 54.45 | 52.31 | 54.56 |
| FedAvg | 60.74±0.31 | 54.73±0.50 | 56.36±0.39 | 65.68±0.23 | 58.91±0.22 | 60.33±0.15 | 55.84 | 52.51 | 52.45 | 53.60 |
| FedProx | 42.90±0.24 | 41.15±0.22 | 40.42±0.62 | 47.36±0.38 | 45.46±0.34 | 46.83±0.11 | 45.41 | 44.11 | 43.77 | 44.43 |
| FedPer | 46.61±9.88 | 54.97±13.23 | 44.82±11.61 | 58.38±9.39 | 59.40±9.71 | 62.32±1.56 | 44.22 | 45.51 | 42.67 | 44.13 |
| GCFL | 27.61±2.55 | 19.81±0.57 | 17.53±0.04 | 47.94±0.41 | 45.71±0.25 | 47.47±0.21 | 42.98 | 40.59 | 35.29 | 39.62 |
| FedGNN | 43.10±0.27 | 41.57±0.07 | 40.70±0.74 | 47.55±0.02 | 45.73±0.26 | 47.46±0.25 | 45.97 | 44.85 | 44.69 | 45.17 |
| FedSage+ | **75.06±0.00** | 71.31±0.00 | 69.73±0.00 | 64.95±0.00 | 65.06±0.00 | 59.33±0.00 | 62.37 | 58.99 | 54.83 | 58.73 |
| FED-PUB | 70.88±0.58 | **72.46±0.68** | 65.26±0.59 | 67.71±3.99 | 54.91±0.42 | 62.48±2.92 | 59.20 | 56.11 | 55.30 | 56.87 |
| FedGTA | 33.33±0.51 | 49.97±2.68 | 50.68±3.94 | 53.61±0.36 | 53.79±0.41 | 61.70±0.35 | 47.06 | 53.81 | 52.18 | 51.02 |
| AdaFGL | 34.41±0.63 | 49.82±2.17 | 50.62±4.19 | 54.18±0.45 | 54.87±0.52 | 62.84±0.49 | 49.08 | 55.43 | 53.99 | 52.83 |
| FedIIH (Ours) | 71.09±0.26 | 71.32±0.09 | **70.30±0.10** | **68.32±0.03** | **67.99±0.09** | **65.40±0.07** | **65.26** | **64.64** | **63.66** | **64.52** |

Table 3: Node classification results of different methods on the **heterophilic** graph datasets in the **non-overlapping** subgraph partitioning setting. Accuracy (%) is reported for *Roman-empire* and *Amazon-ratings*, and AUC (%) is reported for *Minesweeper*, *Tolokers*, and *Questions*. The best and second-best results are shown in **bold** and underlined, respectively.

| | Roman-empire | | | Amazon-ratings | | | Minesweeper | | | - |
|---|---|---|---|---|---|---|---|---|---|---|
| Methods | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | - |
| Local | 39.47±0.03 | 34.43±0.14 | 31.28±0.18 | 41.43±0.04 | 41.81±0.14 | 42.57±0.12 | 67.98±0.07 | 64.39±0.10 | 62.73±0.23 | - |
| FedAvg | 40.89±0.25 | 38.66±0.08 | 36.71±0.20 | 39.86±0.06 | 41.40±0.02 | 41.02±0.16 | 69.06±0.07 | 67.95±0.04 | 66.89±0.08 | - |
| FedProx | 36.63±0.14 | 35.31±0.17 | 33.61±0.59 | 37.00±0.00 | 36.60±0.00 | 36.89±0.00 | 68.27±0.05 | 66.75±0.19 | 66.03±0.16 | - |
| FedPer | 23.66±3.27 | 23.27±3.09 | 22.23±3.58 | 32.33±4.23 | 31.58±0.54 | 34.48±2.25 | 61.85±1.02 | 60.13±1.38 | 60.06±3.61 | - |
| GCFL | 37.65±0.27 | 36.32±0.19 | 34.80±0.09 | 37.00±0.00 | 36.60±0.00 | 36.89±0.00 | 68.47±0.06 | 67.13±0.10 | 57.41±12.56 | - |
| FedGNN | 37.46±0.12 | 36.30±0.16 | 34.84±0.13 | 37.00±0.00 | 36.60±0.00 | 36.89±0.00 | 68.47±0.06 | 67.12±0.11 | 66.41±0.23 | - |
| FedSage+ | 57.48±0.00 | 42.55±0.00 | 33.99±0.00 | 36.86±0.00 | 36.71±0.00 | 37.03±0.00 | **76.64±0.00** | **70.56±0.00** | **70.34±0.00** | - |
| FED-PUB | 43.80±0.25 | 40.46±0.16 | 37.73±0.09 | 42.25±0.25 | 42.25±0.06 | **42.88±0.34** | 68.80±0.09 | 67.43±0.25 | 65.98±0.15 | - |
| FedGTA | 59.86±0.04 | 58.32±0.09 | 57.57±0.21 | 40.81±0.24 | 39.44±0.06 | 39.37±0.04 | 54.35±0.73 | 48.20±1.28 | 52.94±1.77 | - |
| AdaFGL | 64.44±0.03 | 61.77±0.02 | 59.55±0.01 | 39.39±0.05 | 41.19±0.15 | 40.71±0.25 | 55.15±0.84 | 50.15±1.63 | 54.18±2.15 | - |
| FedIIH (Ours) | **65.48±0.12** | **63.32±0.06** | **62.42±0.10** | **42.63±0.02** | **42.40±0.05** | 42.65±0.21 | 69.35±0.25 | 68.09±0.26 | 67.37±0.14 | - |
| | Tolokers | | | Questions | | | Avg. | | | |
| Methods | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | 10 Clients | 30 Clients | 50 Clients | All |
| Local | 73.83±0.03 | 69.01±0.31 | 66.63±0.20 | 63.17±0.02 | 57.17±0.08 | 56.13±0.02 | 57.18 | 53.36 | 51.87 | 54.14 |
| FedAvg | 72.99±0.40 | 58.51±0.27 | 55.47±0.42 | 62.80±0.63 | 58.88±0.18 | 60.78±0.27 | 57.12 | 53.08 | 52.17 | 54.12 |
| FedProx | 54.49±1.69 | 45.59±0.41 | 41.49±0.45 | 52.53±0.34 | 51.54±0.41 | 50.72±0.40 | 49.78 | 47.16 | 45.75 | 47.56 |
| FedPer | 39.60±0.11 | 59.44±0.79 | 41.92±0.06 | 61.31±0.29 | 53.41±1.53 | 50.29±0.10 | 43.75 | 45.57 | 41.80 | 43.70 |
| GCFL | 55.91±1.13 | 23.26±0.70 | 18.40±0.25 | 53.04±0.47 | 51.84±0.38 | 51.10±0.38 | 50.41 | 43.03 | 39.72 | 44.39 |
| FedGNN | 56.21±1.20 | 46.85±0.31 | 42.18±0.45 | 53.04±0.47 | 51.86±0.36 | 51.11±0.38 | 50.44 | 47.75 | 46.29 | 48.16 |
| FedSage+ | **74.54±0.00** | 70.88±0.00 | 69.61±0.00 | 64.22±0.00 | 65.34±0.00 | 62.76±0.00 | 61.95 | 57.21 | 54.75 | 57.97 |
| FED-PUB | 74.17±0.29 | 70.35±0.54 | 66.80±0.85 | 65.39±2.44 | 58.38±1.19 | 58.76±0.16 | 58.88 | 55.77 | 54.43 | 56.36 |
| FedGTA | 40.02±1.70 | 47.34±0.75 | 45.81±1.96 | 35.56±5.46 | 50.43±1.05 | 53.33±0.40 | 46.12 | 48.75 | 49.80 | 48.22 |
| AdaFGL | 45.15±2.15 | 49.18±0.84 | 47.54±2.48 | 41.05±6.49 | 52.18±2.16 | 56.46±0.92 | 49.04 | 50.89 | 51.69 | 50.54 |
| FedIIH (Ours) | 71.67±0.02 | **71.69±0.12** | **69.99±0.03** | **68.79±0.09** | **66.98±0.04** | **64.73±0.35** | **63.58** | **62.50** | **61.43** | **62.50** |

Table 4: Node classification results of different methods on the **heterophilic** graph datasets in the **overlapping** subgraph partitioning setting. Accuracy (%) is reported for *Roman-empire* and *Amazon-ratings*, and AUC (%) is reported for *Minesweeper*, *Tolokers*, and *Questions*. The best and second-best results are shown in **bold** and underlined, respectively.

| | Cora | | Amazon-ratings | |
|---|---|---|---|---|
| Methods | non-overlapping | overlapping | non-overlapping | overlapping |
| w/o HM | 78.67±1.17 (↓3.18) | 78.58±0.03 (↓1.99) | 41.69±0.09 (↓2.55) | 38.04±0.09 (↓4.59) |
| w/o VI | 73.51±0.52 (↓8.34) | 78.40±0.25 (↓2.17) | 41.68±0.05 (↓2.56) | 40.70±0.42 (↓1.93) |
| w/o Dis | 78.78±0.63 (↓3.07) | 77.18±0.23 (↓3.39) | 41.20±0.14 (↓3.04) | 39.98±0.06 (↓2.65) |
| FedIIH | **81.85±0.09** | **80.57±0.23** | **44.24±0.10** | **42.63±0.02** |

Table 5: Ablation studies in both non-overlapping and overlapping partitioning settings on two datasets with 10 clients.

## Acknowledgments

This research is supported by NSF of China (Nos: 62336003 and 12371510), and NSF for Distinguished Young Scholar of Jiangsu Province (No: BK20220080).

## References

Arivazhagan, M. G.; Aggarwal, V.; Singh, A. K.; and Choudhary, S. 2019. Federated learning with personalization layers. *arXiv:1912.00818*.

Baek, J.; Jeong, W.; Jin, J.; Yoon, J.; and Hwang, S. J. 2023. Personalized subgraph federated learning. In *International Conference on Machine Learning*, 1396–1415.

Bai, J.; Yu, W.; Xiao, Z.; Havyarimana, V.; Regan, A. C.; Jiang, H.; and Jiao, L. 2022. Two-stream spatial–temporal graph convolutional networks for driver drowsiness detection. *IEEE Transactions on Cybernetics*, 52(12): 13821–13833.

Chen, F.; Luo, M.; Dong, Z.; Li, Z.; and He, X. 2018. Federated meta-learning with fast convergence and efficient communication. *arXiv:1802.07876*.

Fallah, A.; Mokhtari, A.; and Ozdaglar, A. 2020. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In *Advances in Neural Information Processing Systems*, 3557–3568.

Gelman, A.; and Hill, J. 2006. *Data analysis using regression and multilevel/hierarchical models*. Cambridge University Press.

Guo, J.; Huang, K.; Yi, X.; and Zhang, R. 2024. Learning disentangled graph convolutional networks locally and globally. *IEEE Transactions on Neural Networks and Learning Systems*, 35(3): 3640–3651.

Huang, W.; Wan, G.; Ye, M.; and Du, B. 2023. Federated graph semantic and structural learning. In *International Joint Conference on Artificial Intelligence*, 1–9.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational bayes. *arXiv:1312.6114*.

Kipf, T. N.; and Welling, M. 2016. Variational graph auto-encoders. *arXiv:1611.07308*.

Li, H.; Wang, X.; Zhang, Z.; Yuan, Z.; Li, H.; and Zhu, W. 2021. Disentangled contrastive learning on graphs. In *Advances in Neural Information Processing Systems*, 21872–21884.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems*, 429–450.

Li, X.; Wu, Z.; Zhang, W.; Zhu, Y.; Li, R.-H.; and Wang, G. 2023. FedGTA: Topology-aware averaging for federated graph learning. In *Proceedings of the VLDB Endowment*, 41–50.

Ma, J.; Cui, P.; Kuang, K.; Wang, X.; and Zhu, W. 2019. Disentangled graph convolutional networks. In *International Conference on Machine Learning*, 4212–4221.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *International Conference on Artificial Intelligence and Statistics*, 1273–1282.

Moreno, P.; Ho, P.; and Vasconcelos, N. 2003. A Kullback-Leibler divergence based kernel for SVM classification in multimedia applications. In *Advances in Neural Information Processing Systems*, 1–8.

Pillutla, K.; Malik, K.; Mohamed, A.-R.; Rabbat, M.; Sanjabi, M.; and Xiao, L. 2022. Federated learning with partial model personalization. In *International Conference on Machine Learning*, 17716–17758.

Platonov, O.; Kuznedelev, D.; Diskin, M.; Babenko, A.; and Prokhorenkova, L. 2023. A critical look at the evaluation of GNNs under heterophily: Are we really making progress? In *International Conference on Learning Representations*, 1–15.

Sutter, T.; Daunhawer, I.; and Vogt, J. 2020. Multimodal generative learning utilizing Jensen-Shannon-Divergence. In *Advances in Neural Information Processing Systems*, 6100–6110.

T. Dinh, C.; Tran, N.; and Nguyen, J. 2020. Personalized federated learning with moreau envelopes. In *Advances in Neural Information Processing Systems*, 21394–21405.

Tan, A. Z.; Yu, H.; Cui, L.; and Yang, Q. 2023a. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(12): 9587–9603.

Tan, Y.; Liu, Y.; Long, G.; Jiang, J.; Lu, Q.; and Zhang, C. 2023b. Federated learning on non-iid graphs via structural knowledge sharing. In *AAAI Conference on Artificial Intelligence*, 1–9.

Tran, D.; Ranganath, R.; and Blei, D. 2017. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, 1–11.

Tu, W.; Liao, Q.; Zhou, S.; Peng, X.; Ma, C.; Liu, Z.; Liu, X.; Cai, Z.; and He, K. 2024a. RARE: Robust masked graph autoencoder. *IEEE Transactions on Knowledge and Data Engineering*, 36(10): 5340–5353.

Tu, W.; Zhou, S.; Liu, X.; Ge, C.; Cai, Z.; and Liu, Y. 2024b. Hierarchically contrastive hard sample mining for graph self-supervised pre-training. *IEEE Transactions on Neural Networks and Learning Systems*, 35(11): 16748–16761.

Tu, W.; Zhou, S.; Liu, X.; Guo, X.; Cai, Z.; Zhu, E.; and Cheng, J. 2021. Deep fusion clustering network. In *AAAI Conference on Artificial Intelligence*, 9978–9987.

Xie, H.; Ma, J.; Xiong, L.; and Yang, C. 2021. Federated graph classification over non-iid graphs. In *Advances in Neural Information Processing Systems*, 18839–18852.

Yu, W.; Chen, S.; Gong, C.; Niu, G.; and Sugiyama, M. 2023a. Atom-motif contrastive transformer for molecular property prediction. *arXiv preprint arXiv:2310.07351*.

Yu, W.; Wan, S.; Li, G.; Yang, J.; and Gong, C. 2023b. Hyperspectral image classification with contrastive graph convolutional network. *IEEE Transactions on Geoscience and Remote Sensing*, 61(1): 1–15.

Zhang, C.; Bütepage, J.; Kjellström, H.; and Mandt, S. 2019. Advances in variational inference. *IEEE Transactions on*

*Pattern Analysis and Machine Intelligence*, 41(8): 2008–2026.

Zhang, J.; Hua, Y.; Wang, H.; Song, T.; Xue, Z.; Ma, R.; and Guan, H. 2023. FedALA: Adaptive local aggregation for personalized federated learning. In *AAAI Conference on Artificial Intelligence*, 1–8.

Zhang, Z.; Hu, Q.; Yu, Y.; Gao, W.; and Liu, Q. 2024. FedGT: Federated node classification with scalable graph transformer. *arXiv:2401.15203*.

Zhou, H.; Yu, W.; Wan, S.; Tong, Y.; Gu, T.; and Gong, C. 2024. Traffic pattern sharing for federated traffic flow prediction with personalization. In *International Conference on Data Mining*, 1–10.