# Federated Graph Learning under Domain Shift with Generalizable Prototypes

## Guancheng Wan, Wenke Huang, Mang Ye*

National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence
Hubei Key Laboratory of Multimedia and Network Communication Engineering
School of Computer Science, Hubei Luojia Laboratory, Wuhan University, Wuhan, China.
{guanchengwan, wenkehuang, yemang}@whu.edu.cn

## Abstract

Federated Graph Learning is a privacy-preserving collaborative approach for training a shared model on graph-structured data in the distributed environment. However, in real-world scenarios, the client graph data usually originate from diverse domains, this unavoidably hinders the generalization performance of the final global model. To address this challenge, we start the first attempt to investigate this scenario by learning a well-generalizable model. In order to improve the performance of the global model from different perspectives, we propose a novel framework called Federated Graph Learning with Generalizable Prototypes (FGGP). It decouples the global model into two levels and bridges them via prototypes. These prototypes, which are semantic centers derived from the feature extractor, can provide valuable classification information. At the classification model level, we innovatively eschew the traditional classifiers, then instead leverage clustered prototypes to capture fruitful domain information and enhance the discriminative capability of the classes, improving the performance of multi-domain predictions. Furthermore, at the feature extractor level, we go beyond traditional approaches by implicitly injecting distinct global knowledge and employing contrastive learning to obtain more powerful prototypes while enhancing the feature extractor generalization ability. Experimental results on various datasets are presented to validate the effectiveness of the proposed method.

## Introduction

Federated Learning (FL) (Yang et al. 2019; Huang et al. 2023d) is a decentralized machine learning technique that enables multiple parties to collaboratively train a shared model without sharing their private data (Huang et al. 2022; Fang, Ye, and Yang 2023). However, the heterogeneous data from different clients has emerged as a significant issue impacting the performance of the global model, primarily due to divergent optimization directions during local training. While previous works have proposed some solutions to this issue, their applications have been predominantly in the realms of image and natural language processing (Yang, Huang, and Ye 2023). However, this problem is also inevitable when dealing with graph-structured
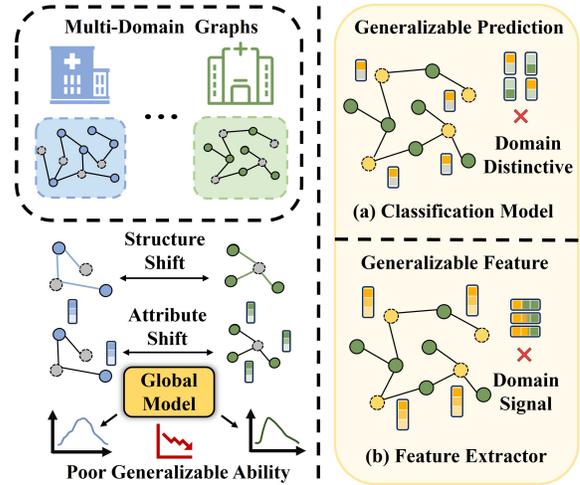


Figure 1: Problem illustration. We illustrate the scenarios of FGL with domain shift. ⬤ ⬤ represents nodes from different domains, ⬤ represents unlabeled nodes, ⬤ is activated by global model in server. Considering attribute and structure shifts, the global model has poor generalization ability, which can be evaluated from two aspects: (a) A singular classifier fails to capture comprehensive information from multi-domain features, thus losing its ability to make generalizable predictions. (b) The feature extractor only generates features in the global direction (yellow) and does not blend signals from different domains (green).

data (Wang et al. 2023; Tian et al. 2023) due to the inherent non-independent and identically distributed (non-IID) nature of graphs. Therefore, a series of Federated Graph Learning (FGL) (Fu et al. 2022; Liu and Yu 2022) methods have been proposed. These methods allow for effective local graph learning while maintaining data privacy, improving the overall performance of the global model.

With the objective of obtaining a better shared global model, existing works in FGL usually achieve collaborative training between subgraphs owners by dealing with missing links (Zhang et al. 2021; Baek et al. 2022), which mainly focused on analyzing distributed graphs within homogeneous domains (Huang et al. 2023a; Yao et al. 2022). However,

this perspective overlooks the complexity of real-world situations where graph data often spans diverse domains (Liu et al. 2023a), leading to the challenging graph domain shift (Huang, Ye, and Du 2022; Huang et al. 2023b). This shift, characterized by disparate distributions of node attributes and graph structures across graph data (Fig. 1), poses greater challenges than the conventional domain shift and hinders the acquisition of a **global model** with **well generalizable ability.** Previous studies on global FGL only considered the relationship between subgraphs within a single domain, thus resulting in limited performance improvement. To address these limitations, our study initiates an innovative exploration of FGL under domain shift with the aim to learn a generalizable global model that can handle multi-graph domain shifts. To the best of our knowledge, we are the first to investigate the impact of domain shift on global FGL. By addressing the unique challenges presented by unknown domain shifts, our research aims to develop relevant benchmarks and make an attempt contribution to this field, potentially shedding light on future investigations in FGL.

The presence of domain shifts in graph data introduces substantial heterogeneity, which impedes the generalization ability of the global model. When dealing with heterogeneous data, traditional FL methods achieve better results through some classifier reweighting methods (Xu, Tong, and Huang 2023), while we also observe some work emphasized that classifiers are primarily responsible for the deterioration of global model performance in the face of data heterogeneity (Luo et al. 2021; Li et al. 2023). Considering domain shift, classifiers often tend to overfit to domain-specific features to achieve higher accuracy during local training. However, following global aggregation, the singular classifier fails to capture comprehensive information across features from all domains, thereby losing its capacity for multi-domain prediction. Based on this observation, we raise the question: 1) *how to construct a classification model that captures fruitful domain information, ensuring well generalizable prediction?* Considering the global model, apart from the classifier, the feature extractor plays a crucial role in mapping local data into the feature space. A proficient feature extractor can be viewed as effectively leveraging local data and acquiring generalizable representations, which in turn can benefit the prediction model. Previous works employ global knowledge as a regularization signal to support local training, providing a global optimization direction for better representation acquisition (Tan et al. 2022a; Mu et al. 2021). However, in the setting of the heterogeneous domain, strict regularization would impede feature generalization due to the fixed global optimization direction, which is harmful to local data exploration with different domains. Moreover, they did not carefully consider the characteristics of graph-structured data and only imposed global knowledge constraints at the feature level. This leads us to consider: 2) *how to achieve agreement between leveraging global knowledge and fully exploring local data, then obtaining generalizable features that benefit the classification model?*

To tackle the challenges mentioned above at two levels, we turn to prototypes and conduct an in-depth exploration of their potential. Prototypes, defined as the mean value of features with identical semantics (Wang et al. 2019), are derived from the feature extractor and can provide diverse classification signals. Thus they serve as a gentle yet powerful bridge between the two levels. In FL, prototypes enhance data privacy, improve robustness against data imbalance, and maintain information across various domains (Tan et al. 2022b). Motivated by these insights, we introduce **F**ederated **G**raph Learning with **G**eneralizable **P**rototypes (FGGP), using prototypes to tackle dual problems. For the classification model level, we introduce Federated Cluster Prototypes Prediction (FCPP). Traditional nonlinear parametric classifiers often suffer from overfitting to specific domains, thereby limiting their ability to capture multi-domain information during aggregation. To mitigate this limitation, we advocate for adopting a prototype-based classification method, which uploads prototypes instead of the classifier after local updates. On the server, these prototypes are clustered, with those in the same domain clustering together. This process retains class information and domain signals. The clustering centers are then stored in a global memory bank. During prediction, we measure distances between samples and these prototypes using the k-nearest neighbors algorithm to infer results.

For the feature extractor level, in order to obtain more generalizable features, we propose Global Knowledge Injected Contrast (GKIC). Prior work utilized prototypes as regularization to constraint local training, differently, our approach involves constructing a novel global view with the support of global knowledge and conducting contrastive learning between this view and the original one. This method not only preserves the model capability to explore local graph data fully but also subtly incorporates global knowledge to facilitate training. Furthermore, we suggest a prototype contrastive learning strategy based on all nodes, optimally utilizing all labeled and unlabeled nodes. This strategy reduces computational overhead, bolsters generalization, and generates representative and semantically rich prototypes, which can effectively serve our proposed classification model. We conjecture that these two components together make FGGP a competitive method for FGL with domain shift. The main contributions are summarized below.

- We are the pioneers in FGL to probe the impact of domain shift with the objective of acquiring a well generalizable global model. In real-world applications, graphs in FGL are often domain-diverse, constituting a more challenging environment. Our research endeavors to create relative benchmarks and contribute to the field, providing valuable insights for future research in FGL.

- We suggest a new method that separates the model into two components, linking them with prototypes. This technique uses clustering prototypes to enhance multi-domain prediction in classification models. For feature extractor, it applies contrastive learning for local and global consistency, achieving better representations and ensuring prototype effectiveness.

- We have performed extensive experiments on a variety of graph datasets exhibiting domain shifts. When contrasted with prior work in FL, our approach effectively mitigates the issues related to domain shift. The result verifies the validity of our proposed method.

## Related Work

### Federated Graph Learning

Federated Graph Learning (FGL) applies the principles of federated learning to graph neural networks, enabling collaborative training on graph-structured data while preserving data privacy (Tan et al. 2023b). FGL allows multiple clients to jointly learn a shared graph representation model without exchanging raw graph data. Previous work primarily focused on establishing a global model on graph datasets within the same domain (He et al. 2021; Chen et al. 2021) or suggesting personalized models across different graph datasets (Baek et al. 2022; Gauthier et al. 2023). However, in real-world scenarios, graph datasets from different clients may come from different domains (Tan et al. 2023a; Ye et al. 2023), while participants still aspire to build a well-generalizable global model. We are the first to study this issue, decoupling the global model and connecting them through prototypes to investigate the problems of existing FGL work. More importantly, we have fully considered the characteristics of graph data to further enhance local training and improve global model multi-domain generalization performance.

### Prototype Learning

Prototype learning, a subfield of machine learning, focuses on learning a compact representation of the input space. This learning paradigm has shown significant potential in a variety of applications (Li et al. 2021, 2020a). For instance, in supervised classification tasks, the system assigns labels to test images by computing their distance from the prototypes of each class (Snell, Swersky, and Zemel 2017). This approach is deemed to be more robust and stable in managing few-shot and zero-shot scenarios (Tian et al. 2020; Jetley et al. 2015). Furthermore, there has been a growing interest in various areas, *e.g.*, semantic segmentation tasks (Zhou et al. 2022). In the context of federated learning applications, both FedProto (Tan et al. 2022a) and FedProc (Mu et al. 2021) employ prototypes as global knowledge regularization terms within local updates to impose constraints. However, these methods only consider global knowledge as constraints at the feature level, neglecting the exploration of local data, which can lead to degradation in certain scenarios. In this study, we extensively leverage the characteristics of graph data and implicitly incorporate global knowledge, rather than using it solely as a regularization factor. Specifically, we construct a global view supported by global knowledge from both semantic and structural perspectives, which is then utilized to achieve consistency between global knowledge and local data, ultimately establishing a more powerful prototype through contrastive learning.

### Contrastive Learning on Graph

Contrastive learning (Ye et al. 2022, 2019), a methodology discerning similarities and disparities within datasets, has demonstrated significant potential when employed in graph data analysis (Liu et al. 2022a, 2021). Initial works such as DGI (Velickovic et al. 2019) and InfoGraph (Sun et al. 2020) maximized the mutual information between graph-level and substructure-level representations. Similar to SimCLR (Chen et al. 2020), GRACE (Zhu et al. 2020) maximizes the agreement of node embeddings across two corrupted views of the graph. More recently, SimGRACE (Xia et al. 2022) and BGRL (Thakoor et al. 2021) try to simplify graph contrastive learning by discarding the negatives. GREET (Liu et al. 2023f) combines contrastive learning with heterophilic graphs. (Liu et al. 2023e, 2022b) involve contrastive learning in cluster tasks. In this work, we have given thorough consideration to the setting of FGL, employing global knowledge to facilitate data augmentation. This approach allows us to execute contrastive learning between a globally enriched view and row view, rather than resorting to random graph disruption strategies.

## Methodology

### Preliminaries

In the Federated Graph Learning framework, $M$ participants, each indexed by $m$, possess private graph data $\mathcal{G}_m = (V_m, A_m, X_m)$ and corresponding node labels $Y_m$. Here, $V_m, A_m, X_m$ represent nodes, adjacency matrices, and node features, respectively. In this context of heterogeneous graph data featuring domain shifts, two primary types of distribution shifts are identified:

**Attribute Shift.** Node features $x_v$ for nodes $v \in V$ are sampled based on the corresponding node labels $y_v$. While the distribution of node labels $P(y)$ remains consistent across clients, the conditional distribution of features $P(x|y)$ differs among them. Attribute shift is defined by the condition: $P_m(x|y) \neq P_n(x|y)$, given $P_m(y) = P_n(y)$.

**Structure Shift.** Consider the joint distribution of the adjacency matrix and node labels $P_{A \times Y}$. The structure shift refers to $P_m(A, Y) \neq P_n(A, Y) (P_m(y) = P_n(y))$. We further explain attribute shift and structure shift in Fig. 1.

Additionally, participants have agreed to share a model with a uniform architecture. The model is composed of two key modules: a feature extractor and a classification model. The feature extractor, symbolized by $f$, often utilizes Graph Neural Networks (GNN), such as Graph Convolutional Networks (GCN) (Kipf and Welling 2017). This extractor encodes a local graph $(X, A)$ into a compact $d$-dimensional feature vector, $Z = f(X, A) \in \mathbb{R}^{v \times d}$, in the feature space. The classification model, represented by $g$, transforms the feature vector $Z$ into a logits output, $Z \to U \in \mathbb{R}^{v \times |I|}$, where $I$ represents the classification categories. The optimization goal is to train a globally generalizable model that performs well across various domains.

### Federated Cluster Prototypes Prediction

**Motivation.** In conventional FGL, the classifier functions as a layer in the model during global aggregation. However, if the graph data from diverse clients demonstrate domain shift, the singular post-global aggregation classifier is only able to map features to logits while neglecting the signals of various domains. This results in subpar generalizable prediction across all domains.

**Prototype-based Prediction** Due to the limited ability of classifiers under highly heterogeneous data, inspired by the non-parameter classifiers in previous works (Wang et al. 2022), we introduce a novel prototype-based classification model. Typical prototype $c^j \in R^d$ is calculated by the mean vector of the features belonging to the same class:

$$c^j = \frac{1}{|\mathbb{N}^j|} \sum_{v_i \in \mathbb{N}^j} Z_i, \qquad (1)$$

where $\mathbb{N}^j$ means the set of nodes annotated with class $j$. Prototypes usually carry the unique semantic information of classes. However, in scenarios of domain shift, the prototypes aggregated from node features also carry domain-specific signals. This encourages us to consider retaining prototypes from different domains globally, then make predictions based on the similarities between node representations and memoried prototypes. Specifically, in the final round of local training, clients upload prototypes to the server memory bank $\mathcal{B}$ instead of classifiers. During the testing phase of the global model on graph data from different domains, we first obtain the representation of the nodes $Z$ through the feature extractor $f$, and then, based on similarity, the top $k$ closed samples in the memory bank are collected to generate a voting prediction. Given a node representation $z_i$, the prediction pipeline can be defined as :

$$\pi_k(z_i) = \text{softmax}( \sum_{j \in \mathcal{I}_{k,\mathcal{B}}(z_i)} \omega(z_i, c_j) \mathbf{1}_{c_j} ), \qquad (2)$$

where $\mathbf{1}_{c_j}$ is the one-hot class label with corresponding prototypes $c_j$, $\mathcal{I}_{k,\mathcal{B}}(z_i)$ donates ordered index of $k$ subset of the global memory bank $\mathcal{B}$, the $\omega(z_i, c_j)$ is the cosine similarity between node representation $z_i$ and prototype $c_j$, which is defined as:

$$\omega(z_i, c_j) = \frac{z_i \cdot c_j}{||z_i|| \times ||c_j||}. \qquad (3)$$

The predictor in Eq. (2) can be viewed as a voting mechanism among the global memory prototypes analogous to the node representation. Initially, we gather the $k$ nearest prototypes to $x_i$ from the memory bank and subsequently compute their cosine similarities. The model will give the label that corresponds to the maximum aggregated weights. In contrast to prior work that directly utilized features as memorized samples (Zhang et al. 2023), employing prototypes offers a more privacy-friendly approach as they have been subjected to a round of aggregation. These prototypes, being discriminative representations drawn from various domains, can provide not only semantic information but also fruitful domain signals. This approach efficiently addresses the issue where the classifier fails to represent multi-domain signals.

**Generalizable Prototypes.** Our initial design of a classification model incorporated domain discrimination abilities. However, it continues to grapple with the following challenges: 1) Given the unique characteristics of graph data learning, nodes may encompass both labeled and unlabeled instances. The conventional computation of prototypes, however, solely incorporates labeled samples, neglecting unlabeled ones. Consequently, the prototypes end up possessing only semantic information skewed toward labeled nodes. 2) As the client count escalates, the global memory bank confronts substantial storage issues, resulting

in a decline in the scalability of this methodology.

In response to the first concern, we procure prototypes by calculating the weighted average of all node representations, the Eq. (1) can be rewritten as follows:

$$c^j = \frac{1}{W^j} \sum_{i \in \arg\max(U_i)=j} w_i Z_i, \qquad (4)$$

where $U$ is the logits output given by local classifier, weight $w_i = max(U_i)$ is the confidence in prediction for unlabeled nodes, otherwise $w_i = 1$ for labeled nodes. $W^j$ is the sum of weight $w_i$ of all nodes allocated to class $j$. In this manner, the prototype effectively capitalizes on all supervision signals. To address the second concern, we introduce a clustering of prototypes. We employ the unsupervised clustering method FINCH (Sarfraz, Sharma, and Stiefelhagen 2019) to derive compact prototype representations within the same domain. Unlike conventional clustering methods, FINCH operates without parameters, making it apt for federated learning with uncertain participant scales and unidentified domains. Given that FINCH clusters through nearest neighbors, prototypes from an identical class but disparate domains have a heightened probability of being clustered into distinct clusters, while those from the same domain are more likely gathered into the same cluster:

$$\mathbb{C}^j = \{c_m^j\}_{m=1}^N \xrightarrow{Cluster} \{c_m^j\}_{m=1}^H \in \mathbb{R}^{H \times d}, \qquad (5)$$

here we assume that the clustering outcome comprises $H$ clusters, with the prototype $\mathbb{C}^j$ of each category $j$ retained in the global memory bank. By doing this, we obtain a compact prototype representation that can characterize a single domain. Concurrently diminishing the complexity from $O(Nd)$ to $O(Hd)$. This modification ensures that the method is no longer constrained by the client count $N$ and therefore enhances scalability.

## Global Knowledge Injected Contrast

**Motivation.** Prior research has considered prototypes as a regularization term of global knowledge in localized training, yet they have solely regarded the constraints of global knowledge, neglecting an exhaustive exploration of local data, leading to these methods not being universally effective. This propels us to consider how to maximize the utilization of local data, while subtly incorporating global knowledge to bolster training, eventually boosting the generalization capability of the feature extractor.

This section describes aligning local graphs with global knowledge using graph contrastive learning. Traditional unsupervised graph studies mainly used random feature masks or edge removal for augmentation. Our approach, however, maintains the original view and adds a global perspective by injecting global knowledge. Recognizing the importance of topology in GNN feature extraction, we reconstruct the graph structure by integrating global knowledge both semantically and structurally.

**Global Semantic K-nearest Neighbor Completion.** Given that the topological connections of local graph data are only present in the original adjacency matrix, they are not connected to potential semantic neighbors, which could help in mining more information during message propagation.

(a) Client: Global Knowledge Injected Contrast      (b) Server: Federated Cluster Prototypes Prediction
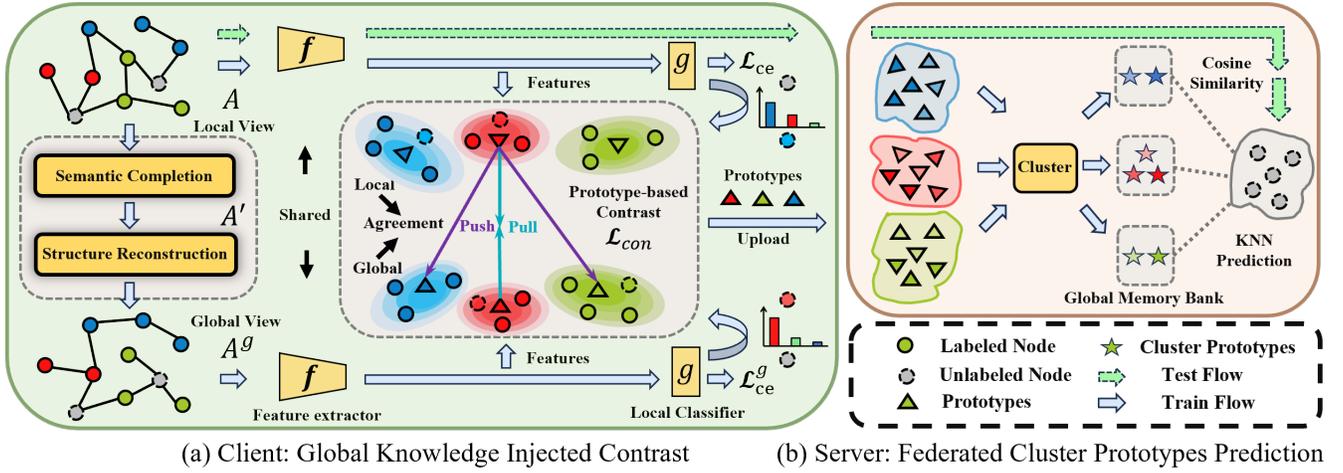
Figure 2: Architecture illustration of Federated Graph Learning with Generalizable Prototypes. We show the two key components in (a) and (b), nodes of different classes are marked with different colors. For cluster prototypes, instances from the same class but different domains are marked with different shades of color. Best viewed in color. Zoom in for details.

Specifically, we preserve the global feature extractor at the local level, which possesses the capability to identify the potential global neighbors of the local nodes. Subsequently, we link each node to its $k'$ semantic neighbors using the $k$-nearest neighbor graph completion (Liu et al. 2023d) method and then incorporate them into the original topological structure. Given the original adjacency matrix $A$ and nodes embeddings $Z'$ given by the global feature extractor $f^g$, this procedure can be expressed as:

$$A'_{ij} = \begin{cases} 1, & \text{if } A_{ij} = 1, \\ 1, & \text{if } i \neq j \text{ and } j \in Top_{k'}(i), \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

Here $Top_{k'}(i)$ means the index of the highest $k'$ similarity nodes to node $i$ measured by cosine similarity distance Eq. (3). Through Eq. (6), we subsequently acquire a new semantically completed view $A'$.

**Global Structure Adaptive Reconstruction.** Prior research highlights the significance of using structural knowledge in graph learning (Liu et al. 2023b, 2022c). We developed an adaptive graph topology augmentation module, $\mathbf{G}$, to integrate global structural data into graph view construction. This module calculates the distribution $\mathcal{P}(A^g|X, A')$ for generating discrete adjacency matrices probabilistically. It first transforms the node feature matrix $X$ into a hidden space representation $F \in \mathbb{R}^{n \times d}$. We then compute a probability matrix $\Omega = \sigma(FF^T) \in \mathbb{R}^{n \times n}$, where $\sigma$ is the sigmoid function and $\Omega_{ij}$ indicates the likelihood of a connection between nodes $v_i$ and $v_j$. To create a discrete adjacency structure and enable differentiable sampling, we use Gumbel-Softmax sampling (Jang, Gu, and Poole 2016; Wu et al. 2022), resulting in a new topology $A^g$:

$$A^g_{i,j} = \lfloor \frac{1}{1 + \exp^{-(\log \mathbf{B}_{i,j} + \mathbf{S})}} + \frac{1}{2} \rfloor, \quad (7)$$

where the $\mathbf{B}_{i,j} := \lambda A'_{ij} + (1-\lambda)\Omega_{ij}, \lambda \in [0, 1]$ stands for the degree of graph augmentation and $\mathbf{S} = \text{Gumbel}(0, 1)$

represents a Gumbel random variate. The module $\mathbf{G}$ is globally shared and aggregated during the training process. This characteristic guarantees that a new topology view can be effectively acquired by leveraging the structural knowledge from various clients.

**Prototype-based Contrastive Learning.** Using both the global and original graph views, we apply contrastive learning across different graphs. Traditional node-level contrastive learning, which neglects existing supervision signals, incurs a high computational cost ($O(n^2)$) on clients. Our approach utilizes a prototype-based method for contrastive learning. We process the views $A^g$ and $A$ using local feature extractors to generate node representations $Z^g$ and $Z$. We then derive prototypes $c^g_j$ and $c_j$ as per Eq. (4). We achieve prototype consistency using the Info-NCE loss (Chen et al. 2020; Liu et al. 2023c):

$$\mathcal{L}_{con} = \frac{-1}{2|I|} \sum_{j=1}^{|I|} \left( \log \frac{s\left(c_j, c^g_j\right)}{\sum_{j \neq l} s\left(c_j, c^g_l\right)} + \log \frac{s\left(c_j, c^g_j\right)}{\sum_{j \neq l} s\left(c_l, c^g_j\right)} \right). \quad (8)$$

Here the $s(c_j, c^g_j) = \exp(\omega(c_j, c^g_j)/\tau)$, $\tau$ is the contrast temperature. By optimizing Eq. (8), we achieve the following advantages: 1) It fully utilizes existing supervision signals, encouraging intra-class compactness and inter-class separability, leading to better node representations. 2) It takes into account unlabeled nodes, preventing nodes from overfitting to labeled nodes through the single CE loss. 3) It combines information from different views, allowing for the local acquisition of more powerful prototypes. 4) Compared to traditional node-level contrastive learning, prototype-level contrastive learning has lower complexity, reducing the computational burden from $O(n^2)$ to $O(|I|^2)$.

## Overall Objective

Finally, we compute the local logit outputs $U$, $U^g$ of the two views and the cross-entropy loss with respect to the la-

| Methods | Twitch | | | | | | | | Citation | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | EN | ES | FR | PT | RU | DE | AVG | △ | DBLP | ACM | AVG | △ |
| FedAvg [ASTAT17] | 58.19 | 48.17 | 44.22 | 49.54 | 36.89 | 65.47 | 50.41 | - | 67.20 | 73.14 | 70.17 | - |
| FedProx [arXiv18] | 58.58 | 40.59 | 42.85 | 45.49 | 33.81 | 66.76 | 48.01 | -2.40 | 67.20 | 73.21 | 70.21 | +0.04 |
| MOON [CVPR21] | 58.75 | 40.54 | 42.73 | 45.23 | 33.81 | 66.76 | 47.97 | -2.44 | 73.38 | 65.50 | 69.44 | -0.73 |
| FedOPT [ICLR21] | 59.42 | 60.22 | 50.52 | 64.05 | 52.85 | 66.58 | 58.94 | +8.53 | 71.73 | 72.98 | 72.35 | +2.18 |
| FedProc [ICLR21] | 58.40 | 46.45 | 45.26 | 48.10 | 38.72 | 66.11 | 50.51 | +0.10 | 66.08 | 73.04 | 69.56 | -0.61 |
| FedSage [NeurIPS21] | 56.40 | 51.56 | 45.17 | 54.20 | 43.51 | 63.02 | 52.31 | +1.90 | 71.01 | 72.71 | 71.86 | +1.69 |
| FedProto [AAAI22] | 58.75 | 41.45 | 42.85 | 45.62 | 34.21 | 66.74 | 48.27 | -2.14 | 66.62 | 73.01 | 69.82 | -0.35 |
| FGGP (ours) | 47.81 | 73.55 | 63.91 | 66.54 | 70.28 | 53.92 | 62.67 | **+12.26** | 82.26 | 67.34 | 74.80 | **+4.63** |

| Methods | Twitch | | | | | | | | Citation | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | EN | ES | FR | PT | RU | DE | AVG | △ | DBLP | ACM | AVG | △ |
| FedAvg [ASTAT17] | 57.21 | 36.40 | 40.33 | 49.29 | 42.22 | 65.37 | 48.47 | - | 78.71 | 70.61 | 74.66 | - |
| FedProx [arXiv18] | 57.52 | 39.46 | 41.17 | 43.53 | 32.50 | 65.82 | 46.67 | -1.80 | 78.74 | 70.97 | 74.86 | +0.20 |
| MOON [CVPR21] | 57.38 | 36.51 | 40.44 | 42.61 | 30.33 | 65.32 | 45.43 | -3.04 | 78.51 | 69.44 | 73.98 | -0.73 |
| FedOPT [ICLR21] | 59.80 | 45.38 | 43.84 | 48.24 | 37.51 | 67.79 | 50.43 | +1.96 | 71.77 | 78.78 | 75.28 | +0.62 |
| FedProc [ICLR21] | 55.70 | 44.19 | 43.53 | 44.44 | 36.49 | 68.50 | 48.81 | +0.37 | 78.58 | 70.34 | 74.46 | -0.20 |
| FedSage [Neurips21] | 59.52 | 43.71 | 43.91 | 47.71 | 36.60 | 67.45 | 49.82 | +1.35 | 78.62 | 72.14 | 75.38 | +0.72 |
| FedProto [AAAI22] | 57.35 | 43.26 | 40.29 | 42.61 | 30.50 | 65.53 | 46.59 | -1.88 | 78.71 | 71.68 | 75.20 | +0.54 |
| FGGP (ours) | 57.49 | 64.25 | 54.29 | 65.10 | 62.66 | 66.63 | 61.74 | **+13.27** | 85.22 | 68.32 | 76.77 | **+2.11** |

Table 1: Comparison with the state-of-the-art methods on Twitch and Citation tasks with $\alpha = 5$ (upper) and $\alpha = 10$ (lower). AVG denotes the average accuracy calculated on all domains. Best in bold and second with underline.

bels $Y$ which is denoted as $L'_{ce} = L_{ce} + L^g_{ce}$. Additionally, to constrain the output of the adaptive graph topology augmentation module, we introduce a regularization term $L_{reg} = \mathbb{H}(A', A^g)$, where $\mathbb{H}$ represents the cross-entropy loss. The overall optimization objective can be expressed as:

$$L = L'_{ce} + \beta L_{con} + \gamma L_{reg}. \tag{9}$$

**Discussion.** We further explain the distinctions between our generalizable prototype and previous applications of prototypes in FL. In previous work, prototypes were primarily used as a regularization technique of global knowledge, aiding in local training. In contrast, our approach focuses on obtaining prototypes that are more generalizable by employing contrastive learning. Additionally, FPL (Huang et al. 2023c) introduces the concept of clustering prototypes, which represents an improvement over previous methods. Yet, these *client-side prototypes* limit local training due to a fixed global optimization direction, constraining generalization, especially across different graph domains. This limitation led us to explore a *server-side* solution. Here, we use cluster prototypes to capture multi-domain signals and class information, enhancing the generality of predictions.

## Experiment

### Experimental Setup

We perform experiments on node-level tasks on graph data with domain shift to confirm the efficacy of FGGP.

**Datasets.** For node classification, our experiment is conducted on three environments under domain shift:

- **Citation.** We follow (Wu et al. 2020) and choose DBLPv8, ACMv9 for different domains, which are obtained from DBLP and ACM respectively. Each node represents a paper and each edge indicates a citation between two papers. The goal is to predict the topic of a paper.
- **Twitch.** Twitch Gamer networks, where nodes represent gamers and edges are followerships between them where different domain represents different language. The task is to predict whether a user streams mature content.

**Network Structure.** Following the common approach in FGL, we utilize GCN as the 2 layers feature extractor $f$ and classifier $g$, with the hidden layer size of 128 for all datasets.

**Implementation Details.** In each experimental environment, we designate the complete graph of each domain as the target for testing on the server side. These graphs are divided into different subgraphs assigned to individual clients, with 60% of nodes used for training. To examine the efficacy of the proposed method under varying degrees of domain shift, we define $\alpha$ as the ratio representing the num of split subgraphs across different domains. An increase in $\alpha$ corresponds to a higher degree of domain shift. In all experimental environments, we set the learning rate to $3e - 4$ and use *SGD* (Robbins and Monro 1951) as the selected optimizer with momentum $0.9$ and weight decay $1e - 5$. The communication round is 200 and the local training epoch is 6 for all datasets. The metric used in our experiments is the node classification accuracy on the testing nodes. We conduct experiments three times and report the last five communication epochs accuracy as the final performance. The code is avail-

(a) Contrastive parameter $\tau$      (b) Neighbor Completion $k'$
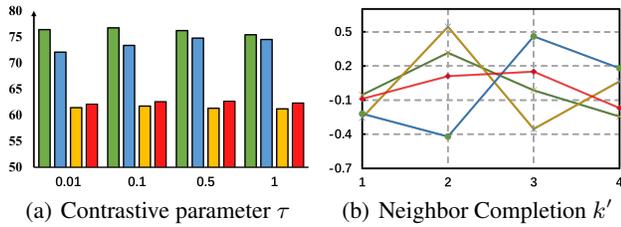
Figure 3: Analysis on hyper-parameter. Performance with hyper-parameter $\tau$ and $k'$, where green and blue represent the Twitch task with $\alpha = 10$ and $\alpha = 5$ respectively, while yellow and red represent the Citation task with $\alpha = 10$ and $\alpha = 5$ respectively.

| FCPP | GKIC | Twitch | | | | | | |
|------|------|------|------|------|------|------|------|------|
| | | EN | ES | FR | PT | RU | DE | AVG |
| ✗ | ✗ | 57.21 | 36.40 | 40.33 | 49.29 | 42.22 | 65.37 | 48.47 |
| ✓ | ✗ | 45.32 | 70.29 | 63.02 | 62.92 | 76.40 | 40.84 | 59.80 |
| ✗ | ✓ | 57.51 | 44.91 | 44.33 | 47.64 | 42.72 | 63.14 | 50.04 |
| ✓ | ✓ | 57.49 | 64.25 | 54.29 | 65.10 | 62.66 | 66.63 | **61.74** |

Table 2: Ablation study of key components in Twitch task.

able at https://github.com/GuanchengWan/FGGP.

**Counterparts.** We compare ours against several SOTA federated approaches focusing on **learning a shared global model**: FedAvg (AISTATS'17 (McMahan et al. 2017)), Fed-Prox (arXiv'18(Li et al. 2020b)), Moon (CVPR'21 (Li, He, and Song 2021)), FedOpt (ICLR'21 (Reddi et al. 2021)), FedProc (arXiv'21 (Mu et al. 2021)), FedSage (NeurIPS'21 (Zhang et al. 2021)) and FedProto (AAAI'22 (Tan et al. 2022a) with parameter averaging).

## Experimental Results

The results of node classification with various state-of-the-art methods under domain shift are presented in Tab. 1. We contrast our approach (FGGP) with several prevalent FL algorithms, including global FL, personal FL, and FGL methods. FGGP effectively enhances multi-domain generalization performance across different environments, achieving higher average performance. When encountering varying degrees of domain shift ($\alpha = 5/10$), the method maintains its competitive performance. We also show the t-SNE results of features on Citation in Fig. 4, which proves that the feature extractor obtains more representative features.

## Diagnostic Analysis

**Hyper-parameter.** We first investigated the $\tau$ in contrastive learning and $k'$ in nearest neighbor completion in the Fig. 3. For $\tau$, within the interval $\tau \in [1, 0.1]$, the impact on performance is not significant. When we set it to a minimum of $\tau = 0.01$, it will cause a certain degree of decline. For $k'$, too few neighbors lead to insufficient semantic completion, and too many will lose the original adjacency information.

| Cluster | Twitch | | | | | | |
|---------|------|------|------|------|------|------|------|
| Method | EN | ES | FR | PT | RU | DE | AVG |
| Kmeans | 43.96 | 71.77 | 63.80 | 70.02 | 71.97 | 53.71 | 62.37 |
| DBSCAN | 44.33 | 69.09 | 65.12 | 64.89 | 76.47 | 48.09 | 61.33 |
| FINCH | 47.81 | 73.55 | 63.91 | 66.54 | 70.28 | 53.92 | **62.67** |

Table 3: Different Cluster Methods Comparison.



(a) DBLP              (b) ACM
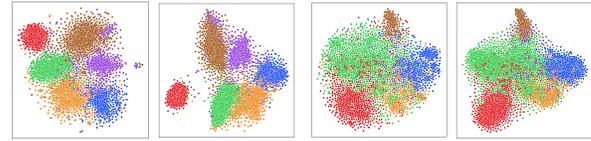
Figure 4: t-SNE Visualization of features in Citation task. In each dataset, the left side represents FedAvg, and the right side represents FGGP.

**The impact of key components.** Tab. 2 discusses two key designs in our method. Firstly, GKIC enhances performance by aligning local and global knowledge. Secondly, FCPP improves the average prediction performance across multiple domains while mitigating the influence of the dominant domain. When these two components are effectively combined, the method achieves its optimal performance.

**Different Cluster Methods.** We compared FINCH with several clustering algorithms (*e.g.*, Kmeans (MacQueen et al. 1967) and DBSCAN (Ester et al. 1996)), and found that these methods require careful hyper-parameter fine-tuning, which makes them less effective in adapting to the FL. Differently, FINCH not only achieves competitive performance but also introduces the advantage of requiring no hyper-parameters. This characteristic enables it to adapt more effectively to the setting where the domains are unknown.

## Conclusion

In this paper, we pioneer an innovative exploration of Federated Graph Learning under Domain Shift with the objective of obtaining a generalizable global model. To achieve this, we decompose the global model into a feature extractor and a classification model, which are connected through prototypes, and propose a novel framework called Federated Graph Learning with Generalizable Prototypes (FGGP). For the classification model, we use Federated Cluster Prototypes Prediction (FCPP) to merge multi-domain data and boost class distinction. At the feature extractor level, our Global Knowledge Injected Contrast (GKIC) aligns local data with global knowledge using contrastive learning, enriching feature diversity and enhancing prototypes. This integrated approach shows promising results and offers insights for advancing Federated Graph Learning.

# Acknowledgments

# References

Baek, J.; Jeong, W.; Jin, J.; Yoon, J.; and Hwang, S. J. 2022. Personalized Subgraph Federated Learning. *arXiv preprint arXiv:2206.10206*.

Chen, F.; Li, P.; Miyazaki, T.; and Wu, C. 2021. Fedgraph: Federated graph learning with intelligent sampling. *IEEE PDS*, 33(8): 1775–1786.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A simple framework for contrastive learning of visual representations. In *ICML*, 1597–1607.

Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.; et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *ACM SIGKDD*, volume 96, 226–231.

Fang, X.; Ye, M.; and Yang, X. 2023. Robust heterogeneous federated learning under data corruption. In *ICCV*, 5020–5030.

Fu, X.; Zhang, B.; Dong, Y.; Chen, C.; and Li, J. 2022. Federated graph machine learning: A survey of concepts, techniques, and applications. *arXiv preprint arXiv:2207.11812*.

Gauthier, F.; Gogineni, V. C.; Werner, S.; Huang, Y.-F.; and Kuh, A. 2023. Personalized Graph Federated Learning with Differential Privacy. *arXiv preprint arXiv:2306.06399*.

He, C.; Balasubramanian, K.; Ceyani, E.; Yang, C.; Xie, H.; Sun, L.; He, L.; Yang, L.; Yu, P. S.; Rong, Y.; et al. 2021. Fedgraphnn: A federated learning system and benchmark for graph neural networks. *arXiv preprint arXiv:2104.07145*.

Huang, W.; Wan, G.; Ye, M.; and Du, B. 2023a. Federated graph semantic and structural learning. In *IJCAI*, 139–143.

Huang, W.; Ye, M.; and Du, B. 2022. Learn from others and be yourself in heterogeneous federated learning. In *CVPR*.

Huang, W.; Ye, M.; Du, B.; and Gao, X. 2022. Few-Shot Model Agnostic Federated Learning. In *ACM MM*, 7309–7316.

Huang, W.; Ye, M.; Shi, Z.; and Du, B. 2023b. Generalizable heterogeneous federated cross-correlation and instance similarity learning. *IEEE TPAMI*.

Huang, W.; Ye, M.; Shi, Z.; Li, H.; and Du, B. 2023c. Rethinking Federated Learning with Domain Shift: A Prototype View. In *CVPR*.

Huang, W.; Ye, M.; Shi, Z.; Wan, G.; Li, H.; Du, B.; and Yang, Q. 2023d. Federated Learning for Generalization, Robustness, Fairness: A Survey and Benchmark. *arXiv preprint arXiv:2311.06750*.

Jang, E.; Gu, S.; and Poole, B. 2016. Categorical Reparameterization with Gumbel-Softmax. *arXiv e-prints*.

Jetley, S.; Romera-Paredes, B.; Jayasumana, S.; and Torr, P. 2015. Prototypical priors: From improving classification to zero-shot learning. *arXiv preprint arXiv:1512.01192*.

Kipf, T. N.; and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

Li, G.; Jampani, V.; Sevilla-Lara, L.; Sun, D.; Kim, J.; and Kim, J. 2021. Adaptive prototype learning and allocation for few-shot segmentation. In *CVPR*, 8334–8343.

Li, J.; Zhou, P.; Xiong, C.; and Hoi, S. C. 2020a. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*.

Li, Q.; He, B.; and Song, D. 2021. Model-Contrastive Federated Learning. In *CVPR*, 10713–10722.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020b. Federated Optimization in Heterogeneous Networks. *arXiv preprint arXiv:1812.06127*.

Li, Z.; Shang, X.; He, R.; Lin, T.; and Wu, C. 2023. No Fear of Classifier Biases: Neural Collapse Inspired Federated Learning with Synthetic and Fixed Classifier. *arXiv preprint arXiv:2303.10058*.

Liu, R.; and Yu, H. 2022. Federated Graph Neural Networks: Overview, Techniques and Challenges. *arXiv preprint arXiv:2202.07256*.

Liu, S.; Li, T.; Feng, Y.; Tran, N.; Zhao, H.; Qiu, Q.; and Li, P. 2023a. Structural Re-weighting Improves Graph Domain Adaptation. In *ICML*, 21778–21793. PMLR.

Liu, Y.; Ding, K.; Liu, H.; and Pan, S. 2023b. GOOD-D: On Unsupervised Graph Out-Of-Distribution Detection. In *WSDM*.

Liu, Y.; Ding, K.; Lu, Q.; Li, F.; Zhang, L. Y.; and Pan, S. 2023c. Towards Self-Interpretable Graph-Level Anomaly Detection. In *NeurIPS*.

Liu, Y.; Ding, K.; Wang, J.; Lee, V.; Liu, H.; and Pan, S. 2023d. Learning Strong Graph Neural Networks with Weak Information. In *ACM SIGKDD*.

Liu, Y.; Jin, M.; Pan, S.; Zhou, C.; Zheng, Y.; Xia, F.; and Yu, P. 2022a. Graph self-supervised learning: A survey. *IEEE TKDE*.

Liu, Y.; Li, Z.; Pan, S.; Gong, C.; Zhou, C.; and Karypis, G. 2021. Anomaly Detection on Attributed Networks via Contrastive Self-Supervised Learning. *IEEE TNNLS*.

Liu, Y.; Liang, K.; Xia, J.; Zhou, S.; Yang, X.; Liu, X.; and Li, S. Z. 2023e. Dink-net: Neural clustering on large graphs. In *ICML*.

Liu, Y.; Xia, J.; Zhou, S.; Wang, S.; Guo, X.; Yang, X.; Liang, K.; Tu, W.; Li, Z. S.; and Liu, X. 2022b. A Survey of Deep Graph Clustering: Taxonomy, Challenge, and Application. *arXiv preprint arXiv:2211.12875*.

Liu, Y.; Zheng, Y.; Zhang, D.; Chen, H.; Peng, H.; and Pan, S. 2022c. Towards unsupervised deep graph structure learning. In *WWW*, 1392–1403.

Liu, Y.; Zheng, Y.; Zhang, D.; Lee, V.; and Pan, S. 2023f. Beyond Smoothing: Unsupervised Graph Representation Learning with Edge Heterophily Discriminating. In *AAAI*.

Luo, M.; Chen, F.; Hu, D.; Zhang, Y.; Liang, J.; and Feng, J. 2021. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. In *NeurIPS*.

MacQueen, J.; et al. 1967. Some methods for classification and analysis of multivariate observations. In *BSMSP*, 281–297.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *AISTATS*, 1273–1282.

Mu, X.; Shen, Y.; Cheng, K.; Geng, X.; Fu, J.; Zhang, T.; and Zhang, Z. 2021. FedProc: Prototypical Contrastive Federated Learning on Non-IID data. *arXiv preprint arXiv:2109.12273*.

Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečný, J.; Kumar, S.; and McMahan, H. B. 2021. Adaptive federated optimization. In *ICLR*.

Robbins, H.; and Monro, S. 1951. A stochastic approximation method. *AoMS*, 400–407.

Sarfraz, M. S.; Sharma, V.; and Stiefelhagen, R. 2019. Efficient Parameter-free Clustering Using First Neighbor Relations. In *CVPR*, 8934–8943.

Snell, J.; Swersky, K.; and Zemel, R. 2017. Prototypical networks for few-shot learning. *NeurIPS*, 30.

Sun, F.-Y.; Hoffmann, J.; Verma, V.; and Tang, J. 2020. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *ICLR*.

Tan, Y.; Chen, C.; Zhuang, W.; Dong, X.; Lyu, L.; and Long, G. 2023a. Is heterogeneity notorious? taming heterogeneity to handle test-time shift in federated learning. In *NeurIPS*.

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*, volume 37, 9953–9961.

Tan, Y.; Long, G.; Liu, L.; Zhou, T.; Lu, Q.; Jiang, J.; and Zhang, C. 2022a. FedProto: Federated Prototype Learning across Heterogeneous Clients. In *AAAI*.

Tan, Y.; Long, G.; Ma, J.; Liu, L.; Zhou, T.; and Jiang, J. 2022b. Federated learning from pre-trained models: A contrastive learning approach. *NeurIPS*, 35: 19332–19344.

Thakoor, S.; Tallec, C.; Azar, M. G.; Munos, R.; Veličković, P.; and Valko, M. 2021. Bootstrapped representation learning on graphs. In *ICLR Workshop*.

Tian, Y.; Dong, K.; Zhang, C.; Zhang, C.; and Chawla, N. V. 2023. Heterogeneous graph masked autoencoders. In *AAAI*.

Tian, Y.; Wang, Y.; Krishnan, D.; Tenenbaum, J. B.; and Isola, P. 2020. Rethinking few-shot image classification: a good embedding is all you need? In *ECCV*, 266–282. Springer.

Velickovic, P.; Fedus, W.; Hamilton, W. L.; Liò, P.; Bengio, Y.; and Hjelm, R. D. 2019. Deep Graph Infomax. In *ICLR*.

Wang, B.; Zhang, Y.; Shi, J.; Wang, P.; Wang, X.; Bai, L.; and Wang, Y. 2023. Knowledge Expansion and Consolidation for Continual Traffic Prediction With Expanding Graphs. *IEEE TITS*.

Wang, K.; Liew, J. H.; Zou, Y.; Zhou, D.; and Feng, J. 2019. Panet: Few-shot image semantic segmentation with prototype alignment. In *CVPR*, 9197–9206.

Wang, W.; Han, C.; Zhou, T.; and Liu, D. 2022. Visual recognition with deep nearest centroids. *arXiv preprint arXiv:2209.07383*.

Wu, L.; Lin, H.; Huang, Y.; and Li, S. Z. 2022. Knowledge distillation improves graph structure augmentation for graph neural networks. *NeurIPS*, 35: 11815–11827.

Wu, M.; Pan, S.; Zhou, C.; Chang, X.; and Zhu, X. 2020. Unsupervised Domain Adaptive Graph Convolutional Networks. In *WWW*.

Xia, J.; Wu, L.; Chen, J.; Hu, B.; and Li, S. Z. 2022. Simgrace: A simple framework for graph contrastive learning without data augmentation. In *WWW*, 1070–1079.

Xu, J.; Tong, X.; and Huang, S.-L. 2023. Personalized federated learning with feature alignment and classifier collaboration. *arXiv preprint arXiv:2306.11867*.

Yang, Q.; Liu, Y.; Chen, T.; and Tong, Y. 2019. Federated Machine Learning: Concept and Applications. *ACM TIST*, 1–19.

Yang, X.; Huang, W.; and Ye, M. 2023. Dynamic Personalized Federated Learning with Adaptive Differential Privacy. In *NeurIPS*.

Yao, Y.; Jin, W.; Ravi, S.; and Joe-Wong, C. 2022. Fedgcn: Convergence and communication tradeoffs in federated training of graph convolutional networks. *arXiv preprint arXiv:2201.12433*.

Ye, M.; Fang, X.; Du, B.; Yuen, P. C.; and Tao, D. 2023. Heterogeneous Federated Learning: State-of-the-art and Research Challenges. *CSUR*.

Ye, M.; Shen, J.; Zhang, X.; Yuen, P. C.; and Chang, S.-F. 2022. Augmentation Invariant and Instance Spreading Feature for Softmax Embedding. *IEEE TPAMI*, 44(2): 924–939.

Ye, M.; Zhang, X.; Yuen, P. C.; and Chang, S.-F. 2019. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 6210–6219.

Zhang, K.; Yang, C.; Li, X.; Sun, L.; and Yiu, S. M. 2021. Subgraph federated learning with missing neighbor generation. *NeurIPS*, 34: 6671–6682.

Zhang, Y.; Wang, X.; Jin, K.; Yuan, K.; Zhang, Z.; Wang, L.; Jin, R.; and Tan, T. 2023. AdaNPC: Exploring Non-Parametric Classifier for Test-Time Adaptation. In *ICML*, 41647–41676. PMLR.

Zhou, T.; Wang, W.; Konukoglu, E.; and Van Gool, L. 2022. Rethinking semantic segmentation: A prototype view. In *CVPR*, 2582–2593.

Zhu, Y.; Xu, Y.; Yu, F.; Liu, Q.; Wu, S.; and Wang, L. 2020. Deep graph contrastive representation learning. In *ICML*.