DGTF: Cross-Domain Decentralized Graph Learning with Topology-Aware Knowledge Fusion

Ruisheng Zheng[1], Mingyi Li[2], Xiao Zhang[2*], Hongjian Shi[2], Yanjie Fu[3], Yuan Yuan[4,5], Dongxiao Yu[2]

[1]School of Cyber Science and Technology, Shandong University
[2]School of Computer Science and Technology, Shandong University
[3]School of Computing and Augmented Intelligence, Arizona State University
[4]School of Software, Shandong University
[5]Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University
{rex7, limee, hongjian}@mail.sdu.edu.cn, {xiaozhang, yyuan, dxyu}@sdu.edu.cn, yanjie.fu@asu.edu

Abstract

Cross-Domain Decentralized Graph Learning (CD-DGL) is a promising paradigm that enables efficient, privacy-preserving collaboration among multiple parties to unlock the value of cross-domain graph data. However, it faces two fundamental challenges. First, inconsistent label spaces across domains drive local models to learn domain-specific biases, which means domain-invariant topological knowledge extraction beyond label constraints is difficult. Second, existing domain topology shift and heterogeneous model architectures make direct model aggregation infeasible. To address these issues, we first use Extended Persistent Homology (EPH) to reveal and quantify the problem of domain topology shift induced by the cross-domain setting. Building on this insight, we present Decentralized Graph Learning with Topology-Aware Knowledge Fusion (DGTF), a novel framework designed to facilitate positive topological knowledge transfer in CD-DGL. Our framework achieves this by integrating two core strategies: first, a contrastive learning-based approach to extract task-agnostic topological knowledge, and second, a topology-aware, model-independent knowledge fusion method to effectively integrate this topological information. Extensive experiments conducted under various cross-domain and model-heterogeneous settings validate the superiority and effectiveness of our proposed framework.

Introduction

Cross-domain graph learning has recently garnered significant attention, aiming to enhance model performance and generalization on a target domain by integrating and leveraging knowledge from multiple graph domains—which typically possess distinct feature spaces, topological properties, and label distributions (Li et al. 2021; Hassani 2022; Liu et al. 2024; Zhao et al. 2024; Yu et al. 2025; He et al. 2025). Concurrently, due to rising privacy concerns in multi-institutional collaborations, cross-domain Federated Graph Learning (CD-FGL) paradigms (Xie et al. 2021) have emerged. This paradigm aims to train optimal graph models for each participant under the coordination of a central server, while preserving the privacy of their respective raw data. Furthermore, to eliminate the reliance on a central coordinator, CD-DGL (He et al. 2022) allows participating nodes to collaborate directly through a peer-to-peer network.

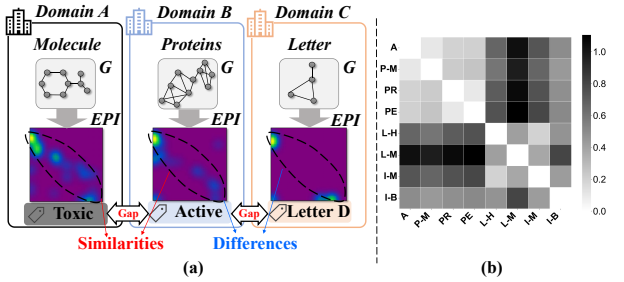

Figure 1: (a) Illustrations of the main challenges in CD-DGL. (b) A heatmap of the pairwise MMD (Maximum Mean Discrepancy) distances between the EPIs generated from eight cross-domain datasets (see Appendix D). The visualization illustrates the topological similarities and differences across these domains.

Existing CD-FGL methods are ill-equipped to cope with the severe data heterogeneity induced by cross-domain settings. A typical case is when one client's domain consists of molecular graphs, while another's is composed of protein interaction networks. Prominent approaches such as GCFL (Xie et al. 2021), FedStar (Tan et al. 2023), and FedSSP (Tan et al. 2024) share cross-domain knowledge that is strongly domain-biased, which inevitably leads to negative knowledge transfer. However, there is still no corresponding work that tackles this problem in the context of CD-DGL. In addition, existing methods exhibit fundamental flaws in resource-heterogeneous scenarios: participants with disparate computational budgets must deploy models of different architectural scales, making the core parameter-aggregation paradigm of these methods infeasible due to this paradigm's rigid assumption of homogeneous architectures. These limitations highlight the urgent need for methods that can support positive knowledge transfer across such diverse and heterogeneous environments.

To achieve cross-domain effective knowledge transfer,

*Xiao Zhang is the corresponding author.

some tricky challenges arise: **(1) Domain-invariant topological knowledge extraction beyond label constraints:** Given that graph data from different domains—such as molecular structures and social networks—possess entirely distinct node attributes and semantics, transferring knowledge based on node features is nearly impossible. Consequently, the only viable foundation for cross-domain knowledge sharing is the extraction of domain-invariant knowledge for which the graph's topology serves as the critical carrier (Mao et al. 2024). However, the training of each local model is strictly guided by its silo-specific downstream tasks and label definitions. This strong local supervision forces the model to capture knowledge with a high **domain bias** tailored specifically to its local objective. For instance, two graphs with identical topology might be labeled identically in one client but differently in another, making the supervised signals themselves domain-specific. This leads to our first critical question: *How can we design a local training paradigm that effectively extracts transferable, domain-invariant topological knowledge while mitigating the influence of biased, domain-specific supervised labels?* **(2) Topological and model heterogeneity:** Even if local models successfully capture valuable information, their aggregation is complicated by two primary issues: topological heterogeneity and model heterogeneity. On one hand, graphs from different domains exhibit significant structural differences, and the degree of underlying commonality varies. Some domains may share considerable topological similarities, while others may be vastly different, as shown in Fig. 1(a). Directly aggregating parameters from models trained on such structurally diverse graphs would likely lead to negative transfer due to conflicting structural patterns. On the other hand, different data silos may employ distinct model architectures to suit their specific needs, rendering a naive, parameter-wise aggregation infeasible. This leads to our second key question: *How can we design an effective aggregation mechanism to fuse heterogeneous models and cross-domain topological knowledge to ensure positive knowledge transfer?*

Along this line, we propose *DGTF*, the first topology-aware, cross-domain, and model-heterogeneous decentralized graph learning framework. First, owing to the advantages of Extended Persistent Homology (EPH) (Cohen-Steiner, Edelsbrunner, and Harer 2009) in graph structure extraction, we process the original graphs into Extended Persistence Images (EPIs) (Adams et al. 2017; Carriere et al. 2020), which contain only topological features. This serves as the foundation for the subsequent topology-aware knowledge transfer. Meanwhile, our experiments (as shown in Fig. 1(b)) verify that commonalities and deviations indeed exist in EPIs across cross-domain graph data, which confirms the presence of topological heterogeneity among these graphs. Subsequently, we introduce a contrastive learning paradigm to learn robust, task-agnostic, graph-level representations locally on each client, which avoids the domain knowledge bias caused by inconsistent label spaces. Furthermore, we share only the parameters of the EPI processing module, so that only domain-invariant topological knowledge is exchanged, while domain-specific knowledge remains encapsulated within each client. We then introduce a cross client attention mechanism to adaptively weigh each client's contribution according to its domain-invariant topological representation. These attention scores guide a multi-teacher distillation module that fuses knowledge across clients, thereby enabling topology-aware knowledge aggregation, mitigating negative transfer, and accommodating model heterogeneity. The main contributions of this paper can be summarized as follows:

- We address the core challenge in Cross-Domain Decentralized Graph Learning (CD-DGL): how to extract and transfer domain-invariant topological knowledge across multiple domain graph data when client models' structures are also heterogeneous.
- We propose *DGTF*, a novel two-stage framework that first learns universal topological structures via self-supervision, and then performs a topology-aware adaptive fusion of this knowledge using a server-free, cross client attention scheme, thereby enabling the positive transfer of topological knowledge across domains while accommodating model heterogeneity.
- We conduct extensive experiments on multiple cross-domain graph datasets with model heterogeneity, demonstrating that our proposed DGTF consistently and significantly outperforms state-of-the-art baselines.

Related Work

Topological GNN

Topological Graph Neural Networks (GNNs) (Pham et al. 2025) integrate features from Topological Data Analysis (TDA) (Wasserman 2018) into GNN architectures, creating highly expressive models for graph representation learning. The first category utilizes traditional Persistent Homology (PH) (Zomorodian and Carlsson 2004; Edelsbrunner, Harer et al. 2008) to capture features like connected components and cycles, such as GFL (Hofer et al. 2021) and TOGL (Horn et al. 2022). A more advanced category leverages Extended Persistent Homology (EPH) (Cohen-Steiner, Edelsbrunner, and Harer 2009). Unlike PH, EPH uses dual ascending and descending filtrations to capture richer pairwise relationships, making it more powerful, as demonstrated in models like PersLay (Carriere et al. 2020) and TopoGCL (Chen, Frias, and Gel 2024). These advances, especially the ability of EPH-based methods to encode rich structural patterns, motivate us to adopt EPH as the backbone for cross-domain, topology-aware knowledge aggregation in our framework.

Federated Graph Learning

Federated Graph Learning (FGL) can be broadly categorized into intra-graph (Baek et al. 2023; Li et al. 2023, 2024a) and inter-graph (Zhu, Luo, and White 2022; Xie et al. 2021) paradigms. In this work, we focus on the inter-graph setting, where each client holds its own set of graphs, typically for graph-level prediction tasks such as graph classification. However, in this federated inter-graph setting, existing methods such as FedStar (Tan et al. 2023) and FedSSP (Tan et al.

2024) struggle in cross-domain scenarios: they tend to propagate domain-specific biases and share conflicting topological information, which hinders positive knowledge transfer. Moreover, model heterogeneity remains largely unaddressed in inter-graph FGL, leaving a clear gap for methods that can simultaneously handle topological shifts and heterogeneous model architectures.

Methodology

Problem Formulation

We consider a CD-DGL setting with a set of N clients, $\mathcal{C} = \{1, \ldots, N\}$, connected in a decentralized, fully-connected topology. Each client $i \in \mathcal{C}$ holds a private graph dataset $\mathcal{D}_i = (G_i, y_i)$, where each graph is denoted as $\mathbf{G}_i = (V_i, E_i)$ with a vertex set V_i and an edge set E_i, and $y_i \in \mathcal{Y}_i$ is the corresponding label. For any two clients $i \neq j$, their graph topologies are non-IID ($P_i(G) \neq P_j(G)$) and their label spaces are inconsistent ($\mathcal{Y}_i \neq \mathcal{Y}_j$). Therefore, we aim to learn a set of heterogeneous local models $\{\theta_i\}_{i\in\mathcal{C}}$ with unique architectures, which makes direct parameter aggregation techniques, such as model averaging, inapplicable. Our objective is to optimize these models collaboratively, enabling each model θ_i to mitigate the domain bias from local supervision while leveraging beneficial topology knowledge from the other clients, $\{\omega_j\}_{j\neq i}$. Here, $\omega \in \theta$ represents a TopSE module that will be detailed in the next part. Formally, each client i solves an optimization problem defined by a local objective function $\mathcal{L}^{(i)}$,

$$\theta_i^* = \arg\min_{\theta_i} \mathcal{L}^{(i)}(\theta_i, \{\omega_j\}_{j\neq i}). \tag{1}$$

Topology-Aware Self-Supervised Learning

Preprocess. To capture the inherent topological features of each graph, while simultaneously preventing the negative transfer caused by node feature heterogeneity in CD-DGL, we first employ an EPH technique to preprocess the graph data. The process begins by constructing a filtration sequence for the graph using K real-valued functions ψ (usually instantiated as graph centrality measures, e.g., degree centrality) defined on the vertex set V. For each function, the construction of the filtration sequence relies on a threshold t. This threshold is a real number used for comparison with the function value of each element in the graph. At any given moment in the filtration process, all elements with a function value ψ less than or equal to the current threshold t constitute the sublevel set, which forms the subgraph for that moment. By continuously varying the value of the threshold t from the minimum to the maximum value of ψ, we obtain a nested sequence of expanding subgraphs formed by these sublevel sets. Specifically, the subgraph G_t is defined as $G_t = \{\sigma \in G \mid \max_{v\in\sigma} \psi(v) \leq t\}$, where σ represents a simplex (e.g., a vertex or edge).

Subsequently, for each filtration sequence, we compute its corresponding Extended Persistence Diagram (EPD). During the computation, the model tracks topological features (such as connected components and cycles) as they evolve through the filtration sequence. Specifically, the model considers two directions of filtration: the sublevel set filtration (with increasing threshold values) and the superlevel set filtration (with decreasing threshold values). It records the filtration value at which each topological feature first appears as its birth time b, and the value at which it disappears or merges with another feature as its death time d. This two-way process ensures that all topological features are assigned a finite birth and death value. Ultimately, for each filtration function, its EPD is defined as the multiset of all tracked features' birth-death pairs (b, d),

$$\text{EPD} = \{(b_f, d_f)\}_{f\in F}, \tag{2}$$

where each (b_f, d_f) is the birth-death pair for a feature.

Since an EPD is a discrete set of points and not suitable for machine learning models, it is converted into a stable, fixed-dimensional matrix representation called an Extended Persistence Image (EPI), following the methodology of (Adams et al. 2017). For clarity, we denote a single EPD as B. The conversion first applies a linear transformation $T(b, d) = (b, d - b)$ to map each point in B from birth-death coordinates to birth-persistence coordinates, forming a new multiset $T(B)$. Next, we generate a persistence surface function ρ_B, defined on the Cartesian plane $\mathbb{R}^2$. Its formal definition is,

$$\rho_B(z) = \sum_{u\in T(B)} f(u)\phi_u(z). \tag{3}$$

In this definition, z is a coordinate variable on the $\mathbb{R}^2$ plane, $f(u)$ is a non-negative weighting function acting on the point u (a common choice is to use the persistence value of the point as the weight), and $\phi_u(z)$ is a probability distribution function with mean u, typically specified as a symmetric Gaussian function $g_u(z)$, which is defined as:

$$g_u(z) = \frac{1}{2\pi\sigma^2} e^{-\|z-u\|^2/2\sigma^2}, \tag{4}$$

where σ^2 is the variance of the Gaussian.

Finally, this continuous surface is discretized by integrating over a predefined $P \times P$ grid (e.g., 50×50) to form the final EPI. The value of each pixel p in the image is given by the following integral:

$$I(B)_p = \iint_p \rho_B(z)\, dydx. \tag{5}$$

This process is repeated for the EPDs generated by each of the K filtration functions, and the resulting K EPIs are ultimately stacked into a tensor $\hat{\mathcal{X}}_{EPI} \in \mathbb{R}^{K\times P\times P}$, which serves as the final topological representation of the graph and can be fed into subsequent models for learning. This multi-filtration approach produces a stable, finite-dimensional representation of the graph's topology. To further enhance the model's robustness against noise, we augment this tensor with Gaussian noise ϵ :

$$\mathcal{X}_{EPI} = \hat{\mathcal{X}}_{EPI} + \epsilon. \tag{6}$$

Topological Representation Module. To extract meaningful patterns from EPI, we introduce a **Topological Structure Encoder (TopSE)**. This module first processes the input tensor $\mathcal{X}_{EPI}$ using a multi-layer convolutional neural

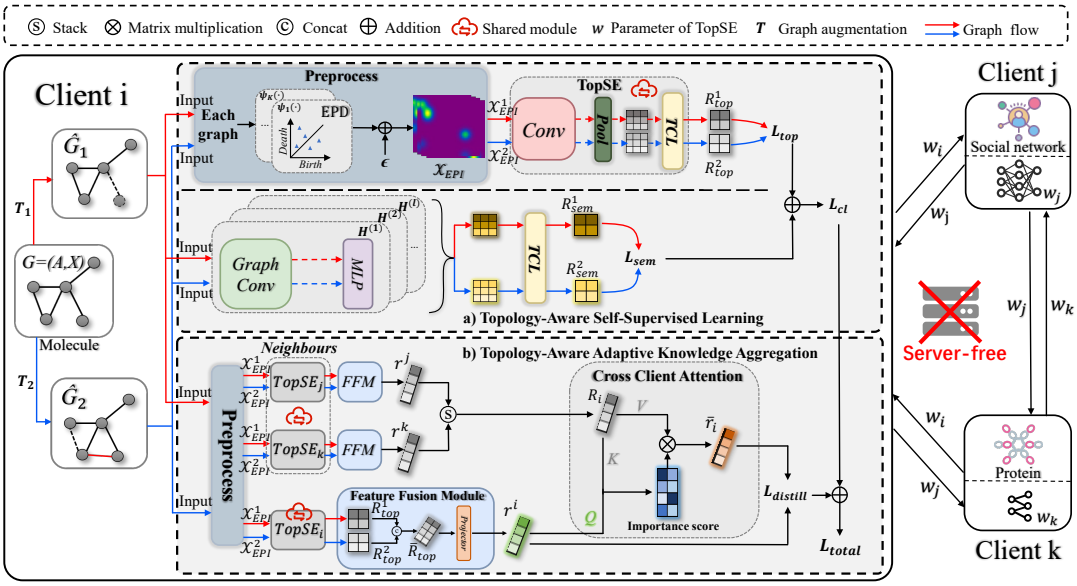

Figure 2: The overall framework of *DGTF* consists of several key components: a) Topology-Aware Self-Supervised Learning: It separately learns domain-invariant topology from EPIs and semantic context from node attributes, using contrastive objectives to alleviate domain bias. b) Topology-Aware Adaptive Knowledge Aggregation: The module adaptively aggregates topological knowledge by leveraging the TopSE from neighboring nodes.

network (Conv). The resulting feature tensor is then passed through a dimension-reduction function $\mathcal{P}$, such as global average pooling. Finally, the result is fed into a Tensor Contraction Layer (TCL) (Kossaifi et al. 2017), which performs a structured compression to efficiently reduce its dimensionality while preserving multi-linear relationships, ultimately producing the compact graph representation $\mathbf{R}_{top}$. The entire transformation is defined as:

$$\mathbf{R}_{top} = \text{TopSE}(\mathcal{X}_{EPI}) = \text{TCL}\left(\mathcal{P}\left(\text{Conv}(\mathcal{X}_{EPI})\right)\right). \quad (7)$$

A key advantage of our approach is that the TopSE module learns domain-invariant topological features, capturing the intrinsic structure of graphs regardless of node feature distributions. Prior work (Adams et al. 2017) supports this, showing that topological features identify underlying patterns in diverse systems despite significant superficial differences. This invariance is especially valuable in heterogeneous decentralized learning, as it provides a robust common ground for knowledge sharing among clients.

In our subsequent decentralized aggregation framework, we strategically share only the parameters associated with the TopSE encoder, facilitating effective transfer of structural knowledge across domains while preserving local specialization for domain-specific node features.

Semantic Context Module. The Semantic Context Module captures multi-scale node semantics through a hierarchical architecture. This module consists of L layers, each containing a Graph Convolution layer followed by an MLP:

$$\mathbf{H}^{(l)} = \text{MLP}^{(l)}(\text{GraphConv}^{(l)}(\mathbf{H}^{(l-1)}, \hat{\mathbf{A}})), \quad (8)$$

where $\mathbf{H}^{(l)}$ represents the node representations at the l-th layer, $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}$ is the normalized adjacency matrix (where $\tilde{\mathbf{A}}$ is the adjacency matrix with self-loops added), and $\mathbf{H}^{(0)}$ is initialized with the input node features. An MLP follows each graph convolution to enhance feature expressiveness, allowing the model to progressively capture semantic context patterns.

Finally, we get a graph-level representation $\mathbf{R}_{sem}$, which is constructed by first concatenating the node representations from each layer and then applying a subsequent transformation layer TCL:

$$\mathbf{R}_{sem} = \text{TCL}\left([\mathbf{H}^{(1)} \parallel \mathbf{H}^{(2)} \parallel \cdots \parallel \mathbf{H}^{(L)}]\right), \quad (9)$$

where $\parallel$ denotes the concatenation operation. This representation provides contextual information complementing an embedding from the Topological Representation Module.

Training Objectives. Local supervised signals, especially from differing label definitions, can lead to domain-biased topological knowledge. This conflicts with the primary goal of extracting universal, domain-invariant patterns. To counteract this bias and enable more effective knowledge transfer, we apply self-supervised learning to both our Topological Representation Module and our Semantic Context Module. Both modules utilize the graph contrastive learning method proposed by (You et al. 2020), which maximizes agreement between different augmented views via a contrastive loss defined as

$$\ell_n = -\log \frac{\exp(\text{sim}(z_{n,i}, z_{n,j})/\tau)}{\sum_{n'=1,n'\neq n}^{N} \exp(\text{sim}(z_{n,i}, z_{n',j})/\tau)}, \quad (10)$$

where $\text{sim}(\cdot,\cdot)$ represents cosine similarity and τ is a temperature parameter. Specifically, in the Topological Representation Module, contrastive learning is applied to the EPIs

generated from augmented graphs, while in the Local Semantic Context Module, contrastive learning is performed directly on the augmented graph. Our framework employs four types of graph augmentations: node dropping, edge perturbation, attribute masking, and subgraph sampling. Different datasets choose different enhancement methods according to their characteristics.

The combined training loss function is formulated as a weighted sum of losses from both modules:

$$\mathcal{L}_{cl} = \mathcal{L}_{sem} + \lambda \mathcal{L}_{top}, \tag{11}$$

where λ is a hyperparameter controlling the contribution of the Topological Representation Module. The final graph representation is obtained by concatenating the representations from both modules:

$$\mathbf{R}_{final} = \text{TCL}\left([\mathbf{R}_{sem} \| \mathbf{R}_{top}]\right), \tag{12}$$

which is then fed into a classification head to generate the final prediction:

$$\hat{\mathbf{y}} = \text{Classifier}(\mathbf{R}_{final}). \tag{13}$$

This self-supervised method directly addresses the issue of disparate label definitions by allowing the model to learn universal structural knowledge from the data itself, independent of domain-specific labels.

Topology-Aware Adaptive Knowledge Aggregation

To address the inherent structural heterogeneity across cross-domain graph data and the failure of direct average aggregation posed by model heterogeneity, we propose a novel topology-aware adaptive multi-teacher knowledge aggregation framework. Our approach enables the preservation of domain-specific knowledge while adaptively extracting domain-invariant topological patterns from other domains.

At the beginning of each communication round, clients exchange their TopSE modules. Specifically, since we consider the fully-connected setting, each client $i \in \mathcal{C}$ interacts with the set of all other clients, which we denote as its neighborhood $\mathcal{N}_i = \{j \in \mathcal{C} \mid j \neq i\}$. Client i transmits its local TopSE module, ω_i, to all clients in $\mathcal{N}_i$. Concurrently, it receives the set of modules $\{\omega_j \mid j \in \mathcal{N}_i\}$ from them. The subsequent representation generation is performed locally on client i. It takes a graph sample G_k from its private dataset $\mathcal{D}_i$ and creates two distinct augmented views. Then, for its own module ω_i and each foreign module ω_j that it has received, client i uses the module to process its own augmented views. This yields pairs of topological representations, denoted as $\mathbf{R}_{\text{top}}^{i,1,k}$ and $\mathbf{R}_{\text{top}}^{i,2,k}$ (from its own module), and $\mathbf{R}_{\text{top}}^{j,1,k}$ and $\mathbf{R}_{\text{top}}^{j,2,k}$ (for each $j \in \mathcal{N}_i$).

Feature Fusion Module. To create a basis for comparison, client i performs two types of concatenations. First, for its own local representations, it computes:

$$\bar{\mathbf{R}}_{\text{top}}^{i,k} = [\mathbf{R}_{\text{top}}^{i,1,k} \parallel \mathbf{R}_{\text{top}}^{i,2,k}]. \tag{14}$$

Second, for each representation pair generated by a neighboring module $j \in \mathcal{N}_i$, it computes:

$$\bar{\mathbf{R}}_{\text{top}}^{j,k} = [\mathbf{R}_{\text{top}}^{j,1,k} \parallel \mathbf{R}_{\text{top}}^{j,2,k}]. \tag{15}$$

Subsequently, client i maps its own representation ($\bar{\mathbf{R}}_{\text{top}}^{i,k}$) and all neighboring representations ($\bar{\mathbf{R}}_{\text{top}}^{j,k}$ for $j \in \mathcal{N}_i$) to an identical low-dimensional space using its private projection head, $Pr_i(\cdot)$:

$$\mathbf{r}^{i,k} = Pr_i(\bar{\mathbf{R}}_{\text{top}}^{i,k}), \quad \mathbf{r}^{j,k} = Pr_i(\bar{\mathbf{R}}_{\text{top}}^{j,k}) \in \mathbb{R}^m, \quad \forall j \in \mathcal{N}_i. \tag{16}$$

Here, m is the dimension of the new latent space. By processing all representations through its unique mapping function, the client obtains topological feature representations with enhanced discriminative capacity.

Cross Client Attention. To facilitate more effective knowledge transfer, we propose a cross client attention mechanism that adaptively learns importance weights for each neighboring client. Specifically, for each graph sample G_k in a local batch $\mathcal{B}_i$, we designate the representation $\mathbf{r}^{i,k}$ as the query. The collection of neighboring representations $\{\mathbf{r}^{j,k} \mid j \in \mathcal{N}_i\}$, generated using neighbor modules on the same sample G_k, serves as both the keys and values. These are stacked into a matrix $\mathbf{R}_{i,k} \in \mathbb{R}^{|\mathcal{N}_i| \times m}$.

The attention-weighted aggregated representation is:

$$\begin{aligned} \bar{\mathbf{r}}_{i,k} &= \text{Attention}(\mathbf{r}^{i,k}, \mathbf{R}_{i,k}) \\ &= \text{softmax}\left(\frac{\mathbf{r}^{i,k} \cdot \mathbf{R}_{i,k}^T}{\sqrt{m}}\right)\mathbf{R}_{i,k}, \end{aligned} \tag{17}$$

where $\bar{\mathbf{r}}_{i,k}$ represents the aggregated cross client topological knowledge for sample G_k.

After obtaining this aggregated representation, we employ a contrastive learning approach where $\bar{\mathbf{r}}_{i,k}$ serves as the positive sample, while representations of other graphs within the same batch $\mathcal{B}_i$ serve as negative samples. The loss is formulated as:

$$\mathcal{L}_{\text{distill}} = -\log \frac{p}{p + \sum_{G_s \in \mathcal{B}_i, s \neq k} p'_{(k,s)}}, \tag{18}$$

where we have $p = \exp(\text{sim}(\mathbf{r}^{i,k}, \bar{\mathbf{r}}_{i,k})/\tau)$ and $p'_{(k,s)} = \exp(\text{sim}(\mathbf{r}^{i,k}, \mathbf{r}^{i,s})/\tau)$. Here, $\mathbf{r}^{i,s}$ is the representation for a negative sample G_s.

Finally, the local loss is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{cl} + \alpha \cdot \mathcal{L}_{\text{distill}}, \tag{19}$$

where α is a hyperparameter.

Experiments

Experimental Setup

Datasets. Following (Tan et al. 2024, 2023), we use 16 public graph classification datasets (Morris et al. 2020) from four domains: Small Molecules (SM) (MUTAG, BZR, COX2, DHFR, PTC_MR, AIDS, NCI1), Bioinformatics (BIO) (KKI, PROTEINS, OHSU, Peking_1), Social Networks (SN) (IMDB-BINARY, IMDB-MULTI), and Computer Vision (CV) (Letter-low, Letter-high, Letter-med). To simulate data heterogeneity, we design four settings of increasing complexity. The first is a cross-dataset setting with the seven SM datasets. The next three are cross-domain settings that progressively combine two (SM-BIO), three (SM-BIO-CV), and finally all four domains (SM-BIO-CV-SN).

Method	SM		SM-BIO		SM-BIO-CV		SM-BIO-CV-SN	
	ACC ↑	F1 ↑	ACC ↑	F1 ↑	ACC ↑	F1 ↑	ACC ↑	F1 ↑
GCN+Local	77.77	70.68	71.84	64.98	70.59	65.50	68.94	63.58
GCN+FML	76.60	63.43	73.61	65.25	71.07	63.68	70.28	63.10
GCN+MH-PFLID	79.65	73.22	72.71	65.41	72.84	66.78	71.07	63.88
GCN+FedMRL	65.49	40.10	64.60	43.34	53.39	36.11	49.30	37.21
GCN+FedType	74.21	60.18	68.03	56.27	64.71	55.52	65.91	55.77
GCN+FedAKT	75.17	60.95	68.75	56.73	66.41	57.40	67.51	57.00
GCFL+Local	76.47	65.15	71.89	62.47	73.98	66.93	73.75	66.99
GCFL+FML	77.06	62.37	71.78	60.78	68.01	60.77	71.91	66.67
GCFL+MH-PFLID	75.71	59.62	72.65	60.36	61.21	47.81	69.00	60.04
GCFL+FedMRL	61.74	44.90	60.87	47.82	55.42	43.32	49.33	33.23
GCFL+FedType	78.23	63.70	71.01	60.81	70.89	61.97	68.79	60.35
GCFL+FedAKT	72.34	59.79	68.16	54.34	67.32	57.45	67.22	60.02
FedStar+Local	77.55	67.69	75.52	68.37	75.15	71.37	72.63	68.54
FedStar+FML	77.93	65.11	71.67	58.40	72.55	63.90	72.63	68.54
FedStar+MH-pFLID	78.46	67.81	73.14	62.84	74.42	66.03	73.37	67.94
FedStar+FedMRL	78.04	70.67	77.74	71.72	77.31	72.12	<u>75.50</u>	70.45
FedStar+FedType	79.31	66.97	72.10	61.04	73.06	64.65	68.99	60.93
FedStar+FedAKT	74.33	60.35	69.02	54.69	67.89	56.50	67.21	56.55
FedSSP+Local	78.45	<u>75.34</u>	73.01	67.46	<u>77.56</u>	<u>71.82</u>	75.41	**71.64**
FedSSP+FML	76.62	66.41	75.84	71.21	74.72	71.75	73.40	70.60
FedSSP+MH-pFLID	<u>80.44</u>	73.85	<u>78.05</u>	<u>74.83</u>	77.14	71.32	74.00	70.11
FedSSP+FedMRL	-	-	-	-	-	-	-	-
FedSSP+FedType	79.78	72.53	72.42	67.80	76.89	70.13	73.92	70.22
FedSSP+FedAKT	-	-	-	-	-	-	-	-
DGTF (ours)	**82.64**	**75.72**	**83.39**	**76.29**	**78.77**	**72.48**	**76.48**	<u>70.65</u>

Table 1: Performance comparison under cross-domain and heterogeneous model settings with best model performance in bold and second-best results with underlines. '-' means this method is not suitable for the experimental setting. In all settings, a client owns a unique dataset.

Baselines. As the first work to jointly address data and model heterogeneity in CD-DGL, we establish comprehensive baselines. We pair SOTA model heterogeneity frameworks (e.g., FML (Shen et al. 2023), MH-pFLID (Xie et al. 2024), FedMRL (Yi et al. 2024), FedType (Wang et al. 2024), FedAKT (Liu et al. 2025)) with two types of client-side graph learners: SOTA cross-domain FGL algorithms such as GCFL (Xie et al. 2021), FedStar (Tan et al. 2023), and FedSSP (Tan et al. 2024), and a standard Graph Convolutional Network (GCN) (Kipf and Welling 2017). For more details on these baselines, please refer to Appendix A.

Implementation Details. We split each client's dataset 90/10 for training/testing. We train for 1 local epoch per round using the Adam (Kingma and Ba 2017) optimizer with a batch size of 64 and learning rate 0.0001. Our method's hyperparameters are set to $\alpha = \{2, 4, 5\}$ and $\lambda = 0.3$. Communication is decentralized over a fully connected topology for 200 rounds. We present the specific architectural configurations for model heterogeneity and other related details in Appendix A.

Evaluation Metrics. We evaluate all methods using their peak Accuracy and Macro F1-score, averaged across the test results of all clients. For evaluation, our contrastive learning method involves freezing the trained encoder and fine-tuning a linear classifier head. In contrast, the fully supervised baselines are trained and evaluated end-to-end.

Experimental Results

Performance Comparison. Table 1 compares the performance of DGTF against baseline methods under model heterogeneity and across four cross-domain non-IID settings. These include one cross-dataset (SM) and three progressive cross-domain scenarios (SM-BIO, SM-BIO-CV, SM-BIO-CV-SN). The results demonstrate that DGTF consistently achieves superior performance over all baselines under varying degrees of domain shift. The results reveal the critical impact of the shared information type on performance. For FGL baselines, combinations involving GCN and GCFL, which share conflicting node features, often underperform their local models. In contrast, topology-only methods like FedStar and FedSSP generally prove superior. This pattern is validated in specific methods: when sharing features, FedType is weaker than FML due to aggravated semantic conflicts, and FedAKT outperforms FedMRL. However, these performance relationships are inverted when only the

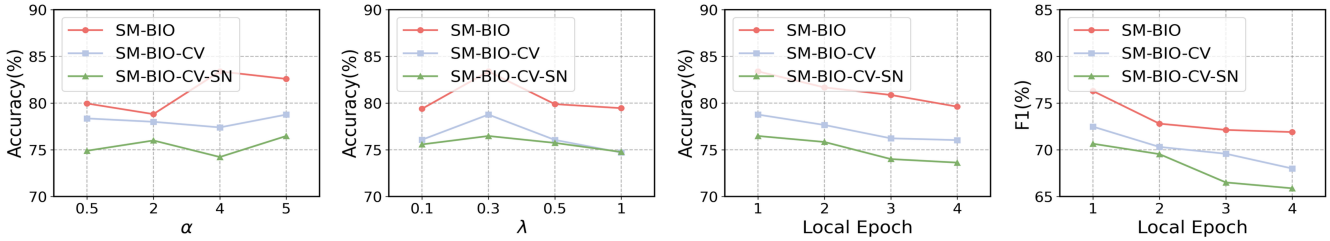

Figure 3: Impact of hyperparameters on model performance under three cross-domain settings, analyzing the effect on Accuracy for α and λ, and on both Accuracy and F1-score for the number of local epochs.

Method	SM-BIO	SM-BIO-CV	SM-BIO-CV-SN
DGTF w/o TRM	75.48	72.72	69.07
DGTF w/o SCM	77.42	62.85	61.92
DGTF w/o CCA	79.97	76.14	73.55
DGTF w/o SSL	76.81	70.79	67.77
DGTF	**83.39**	**78.77**	**76.48**

Table 2: Experimental results on ablation studies of DGTF with best model performance in bold.

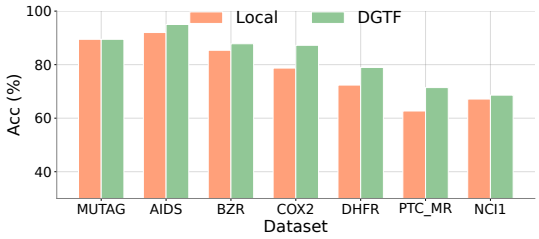

Figure 4: Improvement of client performance, where each dataset represents an individual client.

topology is shared. Among the model heterogeneity baselines, MH-pFLID stands out, effectively mitigating negative knowledge transfer through its attention-based aggregation to achieve the best performance.

Ablation Study. To investigate the effectiveness of different parts of DGTF, we design these comparison experiments:

- DGTF w/o TRM: Only remove the Topological Representation Module (TRM) and share the Semantic Context Module (SCM).
- DGTF w/o SCM: Only remove the SCM.
- DGTF w/o CCA: Only replace the Cross Client Attention mechanism with a uniform weighting scheme for all teacher models.
- DGTF w/o SSL: Only the self-supervised learning of the local models was replaced with standard supervised learning.

Our ablation studies (Table 2) confirm the complementary and indispensable roles of TRM and SCM, as removing either module significantly degraded performance. Replacing our Cross Client Attention with uniform weighting also impaired performance, validating its effectiveness in mitigating knowledge conflicts. Furthermore, switching local training from a self-supervised to a supervised paradigm decreased performance, highlighting self-supervision's crucial role in extracting domain-invariant knowledge for effective aggregation.

Hyper-parameter Study. We conducted a study on key hyperparameters to understand their impact on model performance. The sensitivity analysis for α and λ (Fig. 3) reveals that while both affect performance, the model's sensitivity to their values diminishes as cross-domain heterogeneity increases. Specifically, λ shows a consistent optimal value around 0.3 across all settings, while the optimal α is more sensitive to the degree of heterogeneity, suggesting the model becomes more robust to hyperparameter choices in more complex environments. Furthermore, we investigated the impact of the number of local epochs, as shown in Fig. 3. The results clearly indicate that as local epochs increase, model performance declines across all settings. This trend suggests that excessive local training causes models to overfit to their domain-specific data, hindering the transfer of cross-domain invariant knowledge.

Improvement of clients. Following the setting in (Li et al. 2024b), we compare our method against baseline models trained solely on local data. As presented in Fig. 4, our DGTF framework enhances performance across all clients after collaborative training. This result suggests our framework facilitates positive knowledge transfer among clients.

Conclusion

We presented DGTF, a novel CD-DGL framework that addressed the core challenge of extracting and transferring domain-invariant topological knowledge across multiple graph domains with heterogeneous client model structures. Our approach combined local self-supervised learning for robust topological extraction with topology-aware knowledge distillation for adaptive aggregation. Extensive experiments confirmed DGTF's superiority over state-of-the-art methods in cross-domain and heterogeneous settings.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62202273, in part by the Joint Key Funds of National Natural Science Foundation of China under Grant U24B20149, in part by Major Basic Research Program of Shandong Provincial Natural Science Foundation under Grant ZR2025ZD18, in part by China Postdoctoral Science Foundation under Grant 2024M761806.

References

Adams, H.; Emerson, T.; Kirby, M.; Neville, R.; Peterson, C.; Shipman, P.; Chepushtanova, S.; Hanson, E.; Motta, F.; and Ziegelmeier, L. 2017. Persistence Images: A Stable Vector Representation of Persistent Homology. *Journal of Machine Learning Research*, 18(8): 1–35.

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. PMLR.

Carriere, M.; Chazal, F.; Ike, Y.; Lacombe, T.; Royer, M.; and Umeda, Y. 2020. PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, 2786–2796. PMLR.

Chen, Y.; Frias, J.; and Gel, Y. R. 2024. TopoGCL: Topological Graph Contrastive Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(10): 11453–11461.

Cohen-Steiner, D.; Edelsbrunner, H.; and Harer, J. 2009. Extending Persistence Using Poincaré and Lefschetz Duality. *Foundations of Computational Mathematics*, 9(1): 79–103.

Edelsbrunner, H.; Harer, J.; et al. 2008. Persistent homology-a survey. *Contemporary mathematics*, 453(26): 257–282.

Hassani, K. 2022. Cross-Domain Few-Shot Graph Classification. arXiv:2201.08265.

He, C.; Ceyani, E.; Balasubramanian, K.; Annavaram, M.; and Avestimehr, S. 2022. SpreadGNN: Decentralized Multi-Task Federated Learning for Graph Neural Networks on Molecular Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(6): 6865–6873.

He, Y.; Sui, Y.; He, X.; and Hooi, B. 2025. UniGraph: Learning a Unified Cross-Domain Foundation Model for Text-Attributed Graphs. arXiv:2402.13630.

Hofer, C. D.; Graf, F.; Rieck, B.; Niethammer, M.; and Kwitt, R. 2021. Graph Filtration Learning. arXiv:1905.10996.

Horn, M.; Brouwer, E. D.; Moor, M.; Moreau, Y.; Rieck, B.; and Borgwardt, K. 2022. Topological Graph Neural Networks. arXiv:2102.07835.

Kingma, D. P.; and Ba, J. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. arXiv:1609.02907.

Kossaifi, J.; Khanna, A.; Lipton, Z. C.; Furlanello, T.; and Anandkumar, A. 2017. Tensor Contraction Layers for Parsimonious Deep Nets. arXiv:1706.00439.

Li, A.; Li, Y.; Xue, Z.; Guan, Z.; and Zhuang, M. 2024a. Learning hierarchy-aware federated graph embedding for link prediction. In *2024 IEEE International Conference on Big Data and Smart Computing (BigComp)*, 329–336. IEEE.

Li, M.; Zhang, X.; Wang, Q.; Liu, T.; Wu, R.; Wang, W.; Zhuang, F.; Xiong, H.; and Yu, D. 2024b. Resource-Aware Federated Self-Supervised Learning with Global Class Representations. In Globerson, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J.; and Zhang, C., eds., *Advances in Neural Information Processing Systems*, volume 37, 10008–10035. Curran Associates, Inc.

Li, P.; Wang, J.; Li, Z.; Qiao, Y.; Liu, X.; Ma, F.; Gao, P.; Song, S.; and Xie, G. 2021. Pairwise Half-graph Discrimination: A Simple Graph-level Self-supervised Strategy for Pre-training Graph Neural Networks. arXiv:2110.13567.

Li, X.; Wu, Z.; Zhang, W.; Zhu, Y.; Li, R.-H.; and Wang, G. 2023. FedGTA: Topology-Aware Averaging for Federated Graph Learning. *Proceedings of the VLDB Endowment*, 17(1): 41–50.

Liu, H.; Feng, J.; Kong, L.; Liang, N.; Tao, D.; Chen, Y.; and Zhang, M. 2024. One for All: Towards Training One Graph Model for All Classification Tasks. arXiv:2310.00149.

Liu, S.; Jin, H.; Tang, Z.; Zhai, R.; Lu, K.; Yu, J.; and Bai, C. 2025. Fedakt. *Journal of Systems Architecture*, 159: 103338.

Mao, H.; Chen, Z.; Tang, W.; Zhao, J.; Ma, Y.; Zhao, T.; Shah, N.; Galkin, M.; and Tang, J. 2024. Position: Graph Foundation Models Are Already Here. arXiv:2402.02216.

Morris, C.; Kriege, N. M.; Bause, F.; Kersting, K.; Mutzel, P.; and Neumann, M. 2020. TUDataset: A collection of benchmark datasets for learning with graphs. arXiv:2007.08663.

Pham, P.; Bui, Q.-T.; Nguyen, N. T.; Kozma, R.; Yu, P. S.; and Vo, B. 2025. Topological Data Analysis in Graph Neural Networks: Surveys and Perspectives. *IEEE Transactions on Neural Networks and Learning Systems*, 1–19.

Shen, T.; Zhang, J.; Jia, X.; Zhang, F.; Lv, Z.; Kuang, K.; Wu, C.; and Wu, F. 2023. Federated Mutual Learning: A Collaborative Machine Learning Method for Heterogeneous Data, Models, and Objectives. *Frontiers of Information Technology & Electronic Engineering*, 24(10): 1390–1402.

Tan, Y.; Liu, Y.; Long, G.; Jiang, J.; Lu, Q.; and Zhang, C. 2023. Federated Learning on Non-IID Graphs via Structural Knowledge Sharing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8): 9953–9961.

Tan, Z.; Wan, G.; Huang, W.; and Ye, M. 2024. FedSSP: Federated Graph Learning with Spectral Knowledge and Personalized Preference. *Advances in Neural Information Processing Systems*, 37: 34561–34581.

Wang, J.; Zhao, C.; Lyu, L.; You, Q.; Huai, M.; and Ma, F. 2024. Bridging Model Heterogeneity in Federated Learning via Uncertainty-based Asymmetrical Reciprocity Learning. arXiv:2407.03247.

Wasserman, L. 2018. Topological data analysis. *Annual review of statistics and its application*, 5(2018): 501–532.

Xie, H.; Ma, J.; Xiong, L.; and Yang, C. 2021. Federated Graph Classification over Non-IID Graphs. In *Advances in Neural Information Processing Systems*.

Xie, L.; Lin, M.; Luan, T.; Li, C.; Fang, Y.; Shen, Q.; and Wu, Z. 2024. MH-pFLID: Model Heterogeneous personalized Federated Learning via Injection and Distillation for Medical Data Analysis. arXiv:2405.06822.

Yi, L.; Yu, H.; Ren, C.; Wang, G.; Liu, X.; and Li, X. 2024. Fedmrl. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; and Shen, Y. 2020. Graph Contrastive Learning with Augmentations. In *Advances in Neural Information Processing Systems*, volume 33, 5812–5823. Curran Associates, Inc.

Yu, X.; Gong, Z.; Zhou, C.; Fang, Y.; and Zhang, H. 2025. SAMGPT: Text-free Graph Foundation Model for Multi-domain Pre-training and Cross-domain Adaptation. arXiv:2502.05424.

Zhao, H.; Chen, A.; Sun, X.; Cheng, H.; and Li, J. 2024. All in One and One for All: A Simple yet Effective Method towards Cross-domain Graph Pretraining. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4443–4454.

Zhu, W.; Luo, J.; and White, A. D. 2022. Federated Learning of Molecular Properties with Graph Neural Networks in a Heterogeneous Setting. *Patterns*, 3(6): 100521.

Zomorodian, A.; and Carlsson, G. 2004. Computing persistent homology. In *Proceedings of the twentieth annual symposium on Computational geometry*, 347–356.