FedBG: Proactively Mitigating Bias in Cross-Domain Graph Federated Learning Using Background Data

Sheng Huang, **Lele Fu**, **Tianchi Liao**, **Bowen Deng**, **Chuanfu Zhang** and **Chuan Chen***
Sun Yat-sen University, Guangzhou, China
{huangsh253, fulle, liaotch, dengbw3}@mail2.sysu.edu.cn, {zhangchf9, chenchuan}@mail.sysu.edu.cn

Abstract

Federated graph learning is focused on aggregating knowledge from multi-source graph data and training graph neural networks. Unlike the data that traditional federated learning needs to deal with, federated graph learning also needs to face additional topological information. Further, there are also biases in features and topologies among clients, increasing the difficulty of training models. Previous methods usually seek global calibration information, however, this approach may suffer from information bias caused by data skews, and it is also difficult to naturally combine feature and topology information. Therefore, adjusting the bias before it occurs will hopefully address the learning difficulties caused by the skew. In view of this, we employ background graph data, which works as reference information for local training, to proactively correct bias before it occurs. As a kind of graph data, background graphs are naturally capable of combining feature and topology information to accomplish bias correction among clients in a comprehensive way. Mixing strategy is employed on the background graph to additionally provide privacy-preserving capabilities. Graph generation methods are employed to restore the diversity of background graphs that are blurred by the mixing strategy. Extensive experiments on two real-world datasets demonstrate the sufficient motivation and effectiveness of the proposed method.

1 Introduction

As graph data are generated more and more frequently, there is a growing need for data analysis [Liu *et al.*, 2023; Liu *et al.*, 2025] on them [Zhuang *et al.*, 2024]. Graph neural networks, as a machine learning paradigm for graph data [Yang *et al.*, 2022; Xia *et al.*, 2022], have been used in very many areas where artificial intelligence techniques are needed, e.g., financial analysis [Yang *et al.*, 2021], social recommendation [Liao *et al.*, 2022], bioinformatics [Yi *et al.*, 2022] and so on. However, many existing graph neural network solutions that require centralizing data in one place have difficulty in meeting privacy requirements, due to the huge growth in the amount of data [Hu *et al.*, 2021] and the increasing concern for privacy by many parties [Wang *et al.*, 2022b]. For example, in the case of social media, the exchange of data among different countries may encounter obstacles in terms of policy. As a privacy-secure solution, federated learning can utilize the knowledge provided by more data to build better-performing models without exchanging the original data. Therefore, exploring the combination with federated learning, which enables graph neural networks to access the knowledge provided by more data and obtain more powerful models, is a worthwhile research topic.

Federated Graph Learning (FGL) is a privacy-preserving paradigm for training graph neural networks [Peng *et al.*, 2022; Cai *et al.*, 2024b; Wang *et al.*, 2022a; Cai *et al.*, 2024a]. It improves the performance of the models, by extracting local information on individual clients with the graph data they have and aggregating the information in the communication process. FedAvg [McMahan *et al.*, 2017], as a widely adopted base method for federated learning, can be used for global model aggregation [Qi *et al.*, 2023; Qi *et al.*, 2025], but the problems encountered on traditional federated learning are reproduced on FGL as well. Because of the distributed property of the data, the variability of the optimization objectives in individual clients is widespread in federated learning, and this inconsistency disturbs the generation of federated models [Ye *et al.*, 2023]. And this likewise adversely affects the performance of federated models constructed by FGL. Specifically, the data show different distributions due to their different sources [Zhang *et al.*, 2024c]. For example, social media in different countries may have significantly different exogenous features due to cultural differences, while endogenous friendship preferences may also have differences. In the resulting graph data, such distributional properties may not only be manifested in feature skews, but also in topological variations. And node relationships in the graphs may exist across clients [Guo *et al.*, 2023], so that the propagation paths of some nodes would be lost, and feature skews may result in feature-based composition rules that are not generalizable across clients, which poses a further challenge to FGL.

Similar to previous federated learning methods, the main

* Corresponding author.

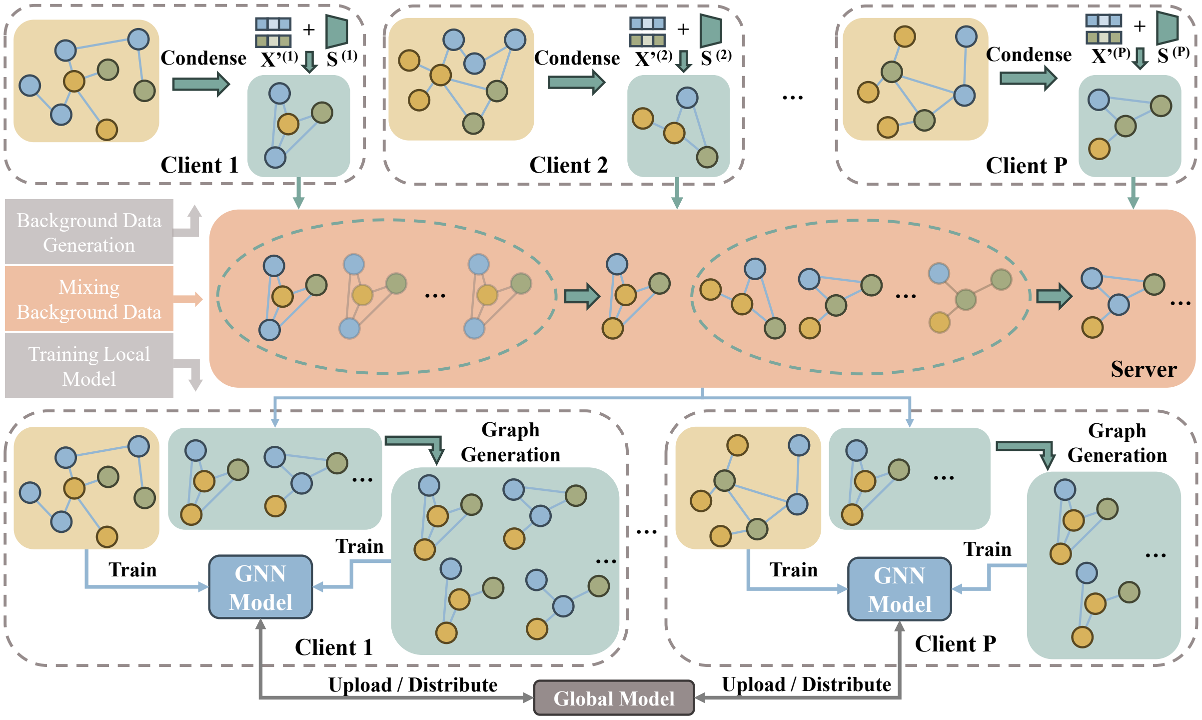


Figure 1: An overview of the proposed FedBG framework. The graph data owned by each client is condensed into background data and then be uploaded to the server, and the mixing operation is performed on the server side to further protect the privacy. The clients use the distributed background data to recover the diversity data before mixing by the graph generation method, and use it as the background for local training to adjust the training process of the local model. The global model is still aggregated from the local models, and the background graph data is intended to provide model tuning before aggregation. They maintain the properties of the graph data and do not need to be dependent on the model being trained (GNN Model).

focus of methods addressing client shifts in graph data is to obtain reference information from other clients and use it to regulate the process of federated training [Liao *et al.*, 2024; Fu *et al.*, 2025]. The simple approach for FGL can be to replace the backbone network in traditional federated learning methods with a graph neural network. Based on this, methods such as FedAvg, FedProx [Li *et al.*, 2020], MOON [Li *et al.*, 2021], etc. can be easily applied to graph data. However, these traditional federated methods ignore an important property in graph data, i.e., topology. Federated methods designed specifically for graph data usually incorporate the consideration of topology, which is learned in order to recover the relationship information between nodes and adjust the training and aggregation process of the models [Xie *et al.*, 2021; Fu and King, 2023; Hu *et al.*, 2024]. Since the graph data are different from client to client, the models obtained by each client are also varied. This type of methods usually adjust the differences only after they have occurred, and predictably such adjustments can lead to a compromise on the knowledge integrity of the models. Further, when dealing with cross-domain graph data, the skewing of feature and topological information among domains would inevitably affect the construction of relationship information between nodes, and would further interfere with the training of the models. In addition, dealing with cross-domain data usually requires the construction of domain calibration information, such as prototypes, which is a common practice in federated learning [Huang *et al.*, 2023b; Zhang *et al.*, 2024a]. In the face of cross-domain graph data, not only feature calibration information, but also topological knowledge needs to be provided for calibration, while combining and utilizing feature and topological information in order to ensure the completeness of the graph data. Moreover, since calibration information is usually obtained and aggregated from models that have already shifted, this leads to model tuning being based on biased calibration information. Therefore, exploring the natural combination of feature and relationship information, and proactively adjust the model before the skew occurs is a way to enhance the performance of cross-domain FGL.

Based on the above analysis, the following objectives need to be achieved: First, in order to construct accurate calibration information, it needs to be extracted directly from the source of information, i.e., the data, rather than indirectly from the model. Second, it is necessary to investigate how to extract common knowledge in cross-domain information in order to counteract the performance damage caused by domain skew. Finally, in order for the provided calibration information to have accurate node relationship information, we need to use calibration information with strong coupling between the feature and topology. Therefore, in this paper, we propose a framework to mitigate the federated graph domain shift problem using background graph data in a proactive manner, called FedBG (**Fed**erated graph learning with **B**ack**G**round data). An overview is shown in Figure 1. Background graphs extracted from the data sources do not rely on models that have been biased, and naturally fused with feature and topology information. A mixing strategy is used to blur the data from individual clients before receiving the background data to further preserve privacy. Training is performed with the background data and the local data to fully utilize all the information, and graph generation techniques are employed to extract common knowledge in the cross-domain data to help FGL counteract the performance degradation caused by domain bias. The contributions of this paper are summarized as follows:

- A method for tuning models on the client side using background data is proposed to mitigate drift proactively before aggregation.
- Employing mixing strategy to blur the background data of individual clients for further privacy protection.
- Utilizing graph diversity generation method to fully extract common knowledge from cross-domain data, and also to defend against performance degradation from mixing strategy.
- Extensive experiments are performed on two real-world datasets, and the results show the advantages of the proposed FedBG over existing state-of-the-art FL and FGL methods.

2 Related Work

2.1 Federated Learning

Federated learning, as a distributed privacy-preserving learning paradigm, has a very high research value in the context of the increasing demand for data privacy protection. FedAvg [McMahan *et al.*, 2017], as a foundational method for federated learning, however, faces the challenge of data heterogeneity. The skewed data distribution tends to negatively affect the effectiveness of federated models, so how to counteract this effect is a key research focus. Limiting the drift caused by heterogeneous data to model training by controlling the updating of model parameters are widely adopted practices [Li *et al.*, 2020; Karimireddy *et al.*, 2020; Wang *et al.*, 2020]. This type of approach are relatively straightforward and therefore have the potential to be improved even further. Using representations generated by the model to guide the updating process can mitigate the information missing due to data heterogeneity [Lin *et al.*, 2020; Li *et al.*, 2021]. This kind of approach improves the performance of federated models by passing information between global and local models. Exploring how to learn about local client data distributions is an interesting solution [Xiong *et al.*, 2023; Huang *et al.*, 2024], which is used to smooth out the heterogeneity of distributions among clients and promote proper model updating. There are many ways of constructing global prototypes to align the representation space of individual clients [Zhang *et al.*, 2024b; Huang *et al.*, 2023b], which can also be used as a federated model building solution for cross-domain data. The above attempts at federated learning to overcome heterogeneity have achieved encouraging success and facilitated the development of federated learning for heterogeneous data. However, these methods do not achieve satisfactory performance in the face of more complex graph data. Since graph data has additional relational information compared with traditional image or text data, how to solve the federated model construction for graph data is a worthy research topic.

2.2 Federated Graph Learning

FGL is a specialized solution for distributed graph data, and unlike traditional federated learning, FGL needs to additionally consider the relationship information between nodes. There are usually two kinds of tasks for graph data, graph-level tasks and node-level tasks, for this paper we focus on the latter. Due to the specificity of graph data, there may also be potential relational links between clients, but due to the inaccessibility of the data, this creates a new challenge for the construction of federated models. Learning the potential links of nodes located between subgraphs of various clients can compensate for the imbalance of data distribution to some extent [Zhang *et al.*, 2021]. Extracting topological information of graph data between clients and using it for cross-client optimization enhancement can boost the performance of graph learning [Baek *et al.*, 2023; Li *et al.*, 2024; Zhu *et al.*, 2024]. Exploring global calibration information among clients can assist in local model training [Wan *et al.*, 2024; Huang *et al.*, 2023a]. Seeking calibration information where features are naturally combined with topological information is beneficial for graph learning. Taking a direct approach from the data side may be able to minimize the interference of bias. Further exploring how to extract common knowledge from cross-domain data can be an effective solution for FGL against domain bias.

3 Methodology

3.1 Preliminaries

In the federated setting, given a graph $\mathcal{G}^{(p)} = (\mathcal{V}^{(p)}, \mathcal{E}^{(p)}, \mathcal{T}^{(p)})$ located on p-th client, where $\mathcal{V}^{(p)}$ denotes a node set, $\mathcal{E}^{(p)}$ denotes the set of edges connected between nodes, and $\mathcal{T}^{(p)}$ is the attributes of the nodes, which includes $\mathbf{X}^{(p)} \in \mathbb{R}^{N^{(p)} \times D}$ and $\mathbf{Y}^{(p)} \in \mathbb{R}^{N^{(p)}}$, indicating the features and labels of each node, respectively. $\mathcal{E}^{(p)}$ can be converted to an affinity matrix $\mathbf{A}^{(p)}$, i.e., $\mathbf{A}^{(p)}_{i,j}$ set to 1 when there is a connected edge between nodes i and j, and 0 otherwise. Based on the graph data $\mathcal{G}^{(p)}$, each client constructs its own graph neural network. Specifically, in the graph node classification task, for the current representation of each node on the graph, which can usually be regarded as the result of message passing, the process can be formalized as the following equation:

$$h_i^{l+1} = \mathrm{UPD}^{(p),l}\left(h_i^l, \mathrm{AGG}\left(\left\{h_j^l : \forall j \in \{j|\mathbf{A}^{(p)}_{i,j} = 1\}\right\}\right)\right), \quad (1)$$

where h_i^l means the representation of i-th node in l-th layer and $h_i^1 = \mathbf{X}_i^{(p)}$. AGG aggregates the neighbor representations of i-th node, and $\mathrm{UPD}^{(p),l}$ updates the l-th layer's representation with the aggregated feature.

For a federated task, we expect all participating clients to jointly construct a global model $\hat{F} = \{\text{UPD}^l\}_{l=1,2,\cdots,L}$ such that it can achieve adaptability over the global data. Therefore, the following loss function needs to be minimized for constructing this federated model:

$$\min_{\hat{\theta}} \mathcal{L}\left(\hat{\theta}; \mathcal{G}\right) = \min_{\hat{\theta}} \sum_{p=1}^{P} \alpha^{(p)} \mathcal{L}^{(p)}\left(\hat{\theta}; \mathcal{G}^{(p)}\right), \tag{2}$$

where $\hat{\theta}$ is the parameter of $\hat{F}$ and $\mathcal{G} = \{\mathcal{G}^{(p)}\}_{p=1,2,\cdots,P}$ means the overall graph data, $\alpha^{(p)}$ means the aggregation weight, which is determined by the number of nodes on the corresponding client. $\mathcal{L}$ denotes the task-related loss function, e.g., the cross-entropy loss.

However, in a distributed environment, there are two critical phenomena that occur in graph data: feature skew and topology skew. **Feature skew:** There exist clients i and j, whose distributions of node features x are different by sampling nodes under the conditions of the same label distribution, i.e., $P_i(x|y) \neq P_j(x|y),\ s.t.\ P_i(y) = P_j(y)$. **Topology skew:** Unlike traditional data, there are also different underlying relationship rules among the graph data at the clients, thus, even with the same feature and label distributions, it may lead to differences in topology, i.e., $P_i(\mathbf{A}|x, y) \neq P_j(\mathbf{A}|x, y),\ s.t.\ P_i(x, y) = P_j(x, y)$ or $P_i(\mathbf{A}|y) \neq P_j(\mathbf{A}|y),\ s.t.\ P_i(y) = P_j(y)$.

The skew among clients leads to dispersion in local model training, which is predictably exacerbated with the cross-domain graph data. Graph data has not only features, but more importantly also relational information, and thus, building a unified global aggregation model for cross-domain graph data is a critical challenge.

3.2 Using Graph Background Data to Provide Calibration Information among Clients

The inability to exchange raw data among different clients makes it difficult for local models to obtain accurate global information as a reference object to regulate the optimization process of local models. At this point, aggregating local models leads to hard calibration, i.e., forcibly merging models with different preferences. As a trade-off between accuracy and privacy, federated learning for general data usually adopts auxiliary representations or model parameters as guidelines for local model tuning. However, since graph data itself has additional relational information, previous approaches have not yet taken this property into account and would ignore this part of the information. Moreover, using imperfect global information as a criterion for local model tuning may lead to a wrong optimization direction. Therefore, considering the characteristics of graph data and exploring the corrective measures before local model aggregation from the original data, it could be expected to solve the phenomenon of client-side shifts caused by graph data skew.

As an intuitive solution, we can calibrate the local model by auxiliary information, which needs to contain both feature and topology information. Moreover, this auxiliary information should be used in a soft way, i.e., not as a constrained way of calibration, so that it is more flexible for the adjustment. Therefore, graph condensation technique is a very suitable solution, which can extract as much feature and structural information as possible from the original graph data to the new graph without exposing the original data. The condensed graphs can be used on the client side to softly tune the local model in order to reduce the bias of individual clients before aggregating. We refer to these new graphs as **background graphs** because they provide the environmental context for federated learning. The node features of the background graph can be constructed using gradient matching. Specifically, for each of the original and background graphs, the gradients $g^{(p)}$ and $g'^{(p)}$ at the model $\theta^{(p)}$ on the p-th client are shown in the following equations, respectively:

$$g^{(p)} = \nabla_{\theta^{(p)}} \mathcal{L}_T^{(p)}\left(\theta^{(p)}; \mathbf{A}^{(p)}, \mathbf{X}^{(p)}, \mathbf{Y}^{(p)}\right), \tag{3}$$

$$g'^{(p)} = \nabla_{\theta^{(p)}} \mathcal{L}_T^{(p)}\left(\theta^{(p)}; \mathbf{A}'^{(p)}, \mathbf{X}'^{(p)}, \mathbf{Y}'^{(p)}\right), \tag{4}$$

where $\mathbf{A}'^{(p)}, \mathbf{X}'^{(p)}$ and $\mathbf{Y}'^{(p)}$ denote the affinity matrix, features and expected output of the background graph, respectively. $\mathcal{L}_T^{(p)}$ means the task loss, e.g. cross-entropy loss. Here, the features of the background graph are set as learnable parameters, while the affinity matrix of the background graph is fixed, then the learnable parameters can be solved by minimizing the following loss function:

$$\mathcal{L}_{GF}^{(p)}(\mathbf{X}'^{(p)}) = D(g^{(p)}, g'^{(p)}), \tag{5}$$

where $D(\cdot, \cdot)$ is a distance function that can be set to Euclidean distance. At this point, the features of the background graph are optimized in a direction that can simulate the optimized properties of the original graph.

The topological construction pattern of the graph as a potential knowledge, which can be acquired by mining the feature relationships between nodes and fitting the client environment in which the nodes are located. Therefore, relational sampler is employed to construct topology information in the current environment:

$$\mathbf{A}'^{(p)}_{i,j} = \sigma\left(\frac{[S_f(F_{i,j}), S_e(i,j)] + [S_f(F_{j,i}), S_e(j,i)]}{2}\right), \tag{6}$$

here, $F_{i,j} = \mathbf{X}'^{(p)}_i || \mathbf{X}'^{(p)}_j$ and "$\cdot||\cdot$" is the concatenation operation, $[\cdot, \cdot]$ means the merge operation, e.g. dot product, σ denotes the Sigmoid function, $S_f = S_f^{(p)}$ denotes the feature relational sampler that focuses on learning the compositional knowledge originating from the features, and $S_e = S_e^{(p)}$ denotes the node relational sampler that focuses on learning the native compositional knowledge originating from the relationships among the nodes, and both of them can be instantiated using multilayer perceptrons. The relational sampler samples from feature space and node space, and generates the topology of the background graph based on the compositional knowledge fused in the model parameters.

Using gradient matching, the constructed topological information is aligned with the original graph in order to learn the knowledge of graph construction in the current environment. At this point, the process is accomplished by minimizing the following loss function:

$$\mathcal{L}_{GT}^{(p)}(\mathbf{A}'^{(p)}) = D(g^{(p)}, g'^{(p)}). \tag{7}$$

The generated background graphs are collected by the server, and distributed to each client for further operations. Towards additional privacy security, the background graphs generated by the individual clients will also be mixed before being distributed. To reduce the resulting performance degradation, cluster mixing is employed. Similar background graphs are mixed to avoid knowledge loss due to conflicting information. Since the topological characterizations are dependent partly on the node features during the generation of the background graphs, the node features of the background graph at each client are therefore adopted as the criterion for the mixing operation. The clustering operation on the collected node features of the background graph can result in the corresponding clustering labels:

$$\mathbf{C} = \text{Cluster}(\{\mathbf{X}'^{(p)}\}_{p=1,2,\cdots,P}), \tag{8}$$

where $\mathbf{C} \in \mathbb{R}^P$ denotes the cluster labels of feature matrices, and intra-cluster fusion of features and the relational sampler parameters is performed using the cluster labels:

$$\mathbf{F}'^{(p)} = \sum_{i \in \{k|\mathbf{C}_k = \mathbf{C}_p\}} \frac{\alpha^{(i)}}{\sum_{j \in \{k|\mathbf{C}_k = \mathbf{C}_p\}} \alpha^{(j)}} \mathbf{F}^{(i)}, \tag{9}$$

where $\mathbf{F}'^{(p)}$ means the the fused node features and the parameters of the relational sampler for the p-th client, and $\mathbf{F}^{(p)} = \{\mathbf{X}'^{(p)}, S_f^{(p)}, S_e^{(p)}\}$ denotes the original data. The fusion result in the new background graph which can be utilized for distribution and used on the client side.

3.3 Background Graph Diversity Adjustment Strategy

After the server completes the collecting and processing of the background graphs, they are distributed to the clients. Based on the received background graphs, each client can perform a client-side tuning process in order to be able to use the global information to provide an optimized reference during local training, and to reduce model differences between clients. There is a necessity to utilize the knowledge of the background graph as fully as possible to explore common knowledge among cross-domain data. And, we note that a mixing strategy is used for the background graphs to preserve privacy, thus, on the client side, additional strategies need to be employed to mitigate the adverse effects of the mixing strategy on client tuning. Inspired by [Wu *et al.*, 2022], we found that for each background graph generated by a client, it can be regarded as the exogenous representations of the compositional endogenous factors generated in a particular environment. Hence, a background graph after fusion should still be able to be extracted with the same number of exogenous representations as the number of environments before fusion. Mining compositional endogenous factors from multiple background graphs helps to train models adapted to a wider range of data among clients and reduces models variance after training. We activate different exogenous representations through graph generation methods to restore the environment in which the original data are generated, and expect local models to adapt to different environments. The process can be implemented using a generative adversarial approach, where a richer set of environments need to be simulated for the graph generator to interfere to the local model, while the local model needs to learn to eliminate the interference and find the accurate compositional endogenous factors. The following optimization objective is adopted to complete the optimization process:

$$\min_{\theta^{(p)}} \max_{\overline{\mathbf{A}}} \text{Var}\{\mathcal{L}_T^{(p)}\left(\theta^{(p)}; \overline{\mathbf{A}}_i, \mathbf{X}'^{(i)}, \mathbf{Y}'^{(i)}\right)\}_{i=\{1,2,\cdots,P\}}, \tag{10}$$

where $\overline{\mathbf{A}}_i = G_i^{(p)}(\mathbf{A}'^{(i)}, \mathbf{X}'^{(i)})$ denotes the i-th generated background graph affinity matrix, which is generated by the graph generator $G_i^{(p)}(\cdot)$, and $\overline{\mathbf{A}}_i \in \overline{\mathbf{A}}$. For local models, it is also necessary to minimize the base loss additionally:

$$\min_{\theta^{(p)}} \frac{1}{P} \sum_{i=1}^{P} \mathcal{L}_T^{(p)}\left(\theta^{(p)}; \overline{\mathbf{A}}_i, \mathbf{X}'^{(i)}, \mathbf{Y}'^{(i)}\right), \tag{11}$$

The calibration of the local model is performed through the adversarial training between the graph generator and the local model. And the above calibration process is performed to assist the following training process for local data:

$$\min_{\theta^{(p)}} \mathcal{L}_T^{(p)}\left(\theta^{(p)}; \mathcal{G}^{(p)}\right). \tag{12}$$

With the auxiliary adjustment effect of the background graphs, the local model is able to accomplish calibration before aggregation and adapt to a wider range of data. The complete algorithm is shown in Appendix.

4 Evaluation

4.1 Datasets and Methods

The evaluations are performed on two real-world datasets, including Twitch and Facebook100. 10% of the nodes of each domain's graph data in Facebook100 are sampled as the training data to make Facebook100-lite. On Twitch dataset, we additionally use the Louvain algorithm to divide the graph of each domain into 2 or 20 subgraphs, in order to simulate a larger number of clients and to boost the difficulty of the federated task. And, we also provide the experimental results under another setup.

We perform the comparative experiments with seven federated learning methods, including three traditional federated learning methods (FedAvg [McMahan *et al.*, 2017], FedProx [Li *et al.*, 2020] and MOON [Li *et al.*, 2021]), and four federated graph methods (FedPUB [Baek *et al.*, 2023], FGGP [Wan *et al.*, 2024], FedTAD [Zhu *et al.*, 2024] and FedGTA [Li *et al.*, 2024]), for a comprehensive comparison.

4.2 Experimental Results

The results obtained by the proposed FedBG and all the comparison methods on the two real-world datasets are displayed in Table 1 and Table 2. As can be seen from the data in the table, the traditional enhancements applied to FedAvg do not yield as effective a boost on the graph datasets as previous non-graph datasets, which can be considered as a lack of capturing additional relational information on the graph data. In the smaller dataset Twitch, both FGL methods are able to achieve some advantages over traditional FL methods like FedAvg. However, the proposed FedBG is able to

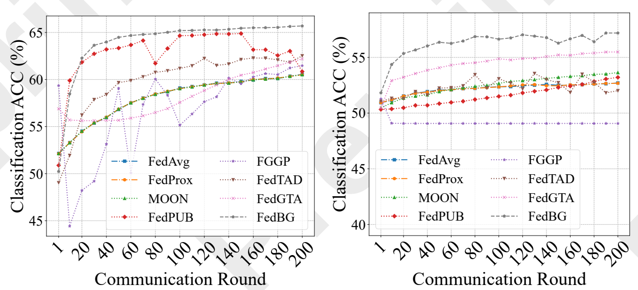
Type	Method	Twitch								Facebook100			
		EN	ES	FR	PT	RU	DE	AVG	Δ	F-AVG	Δ	L-AVG	Δ
FL	FedAvg	56.40	68.87	57.57	66.67	66.59	57.61	60.55	-	54.66	-	52.71	-
	FedProx	56.37	68.76	57.50	66.67	66.48	57.66	60.51	-	54.64	-	52.68	-
	MOON	56.40	68.82	57.54	66.67	66.59	57.53	60.51	-	55.22	-	53.64	-
FGL	FedPUB	45.70	71.45	62.38	67.06	76.05	57.68	60.85	0.30	<u>57.71</u>	3.05	53.21	0.50
	FGGP	60.12	66.34	57.38	64.84	60.83	62.58	61.48	0.93	OOM	-	49.07	-3.64
	FedTAD	56.05	70.38	56.31	67.19	69.10	63.82	<u>62.52</u>	1.97	OOM	-	52.01	-0.70
	FedGTA	50.61	71.61	62.11	69.28	75.66	58.66	62.19	1.64	55.49	0.83	<u>55.49</u>	2.78
FGL	FedBG	59.38	71.99	61.20	69.28	74.34	65.71	**65.69**	5.14	**58.20**	3.54	**57.19**	4.48

Table 1: The results (ACC-%) of two datasets for FedBG and all comparison federated learning methods. **Bold** and <u>underlined</u> results are the best and the second best. F-AVG: Average values over all domains of the Facebook100 dataset. L-AVG: Average values over all domains of the Facebook100-lite dataset.

Type	Method	Twitch								Facebook100	
		EN	ES	FR	PT	RU	DE	AVG	Δ	AVG	Δ
FL	FedAvg	53.53	66.13	58.07	65.36	66.76	58.29	59.81	-	53.64	-
	FedProx	53.56	66.08	58.22	65.49	66.70	58.21	59.81	-	53.62	-
	MOON	53.53	66.13	58.22	65.49	66.82	58.29	59.85	-	53.64	-
FGL	FedPUB	45.88	70.16	62.57	65.36	75.94	59.55	61.15	1.34	54.76	1.12
	FGGP	54.58	65.11	52.80	60.65	56.44	63.11	58.63	-1.18	51.06	-2.58
	FedTAD	54.54	68.76	57.34	65.36	67.33	62.18	<u>61.39</u>	1.58	53.17	-0.47
	FedGTA	48.97	70.48	62.27	67.58	75.20	57.61	61.27	1.46	<u>55.07</u>	1.43
FGL	FedBG	56.23	71.72	60.74	66.14	72.18	64.74	**64.18**	4.37	**57.66**	4.02

Table 2: The results (ACC-%) of different data setup for FedBG and all comparison federated learning methods. **Bold** and <u>underlined</u> results are the best and the second best.

achieve more significant improvements, due to the sufficient and correct utilization of cross-domain graph data by the proposed background graph data and its series of processing methods. There are some methods that expose their high resource requirements in the more difficult and larger dataset Facebook100. After reducing the training cost by random sampling, the FGL methods tend to decline more compared to the FL method because of the large amount of relationship information lost. The proposed method FedBG, on the other hand, works from the data and is able to mitigate the model learning misinterpretation due to data variations with better accuracy. The convergence curves of the training process for all methods are presented in Figure 2. It can be seen that FedBG is able to obtain stable performance improvement with increasing communication rounds in both datasets. On the more difficult dataset Facebook100, the proposed method is also able to obtain stable training results compared to other methods.

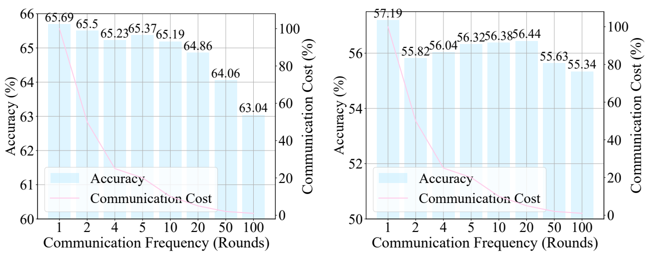


(a) Twitch (b) Facebook100-lite

Figure 2: The classification accuracy curves as the increasing communication round for all comparative methods.

4.3 Communication Cost Reduction

Since the method proposed in this paper uses background graph data as calibration information across domains, and the generation of each background data is done only on the respective client, there is no need to use information from other clients. Therefore, it can be assumed that most of the information carried by the background data is already captured at the time of local generation. Naturally, we believe that the use of background data can be done without the need for the same frequent communication as before. Excellent performances can be obtained by using the background data as reference information for local training. To verify this hypothesis, we set the communication frequency of the background graphs and the network parameters to different values, and the results are shown in Figure 3. It can be seen that the final performance suffers slight degradations due to the reduced communication frequency. However, the performance degradations are insignificant compared to the huge reduction in communication cost. Therefore, FedBG has potential in reducing the communication. The generation of background data does not need to rely on the to-be-trained model with bias. Due to the process of utilizing background information enables the extraction of cross-domain commonality information and calibration of the to-be-trained model to be done locally, there is no longer a need for frequent communication across the clients to obtain more accurate training directions.



(a) Twitch (b) Facebook100-lite

Figure 3: Comparison of classification accuracies and corresponding communication costs of the proposed FedBG for different communication frequencies. FedBG can maintain high classification accuracies even with a significant reduction in the communication cost.

4.4 Ablation Study

By introducing the involved modules in stages, the different results they yield are presented in Figure 4. Since there is a progressive relationship between the modules, the addition of each module is dependent on the previous one. First, the performance of the initial model is significantly increased after providing it with background data, which is the main reason for the effectiveness of the method. Then, in order to improve the privacy, the mixing strategy is employed to blur the background data from different clients. However, this operation impairs the performance, which is particularly noticeable in the Facebook100 dataset. Therefore, graph generation techniques are needed to recover the diversity of background data in order to obtain performance recovery. With the background data diversity restored, we are surprised to find that the performance can even be higher than the model that only uses the original background data. We believe that this graph generation approach is able to additionally introduce more perturbations, which improves the feature extraction capability of the model in a adversarial training. In order to verify whether it is practicable to employ background data training only on the server, we train the federated model by aggregating sufficiently trained background graph data on the server without local data. However, this training approach does not work due to the lack of extensive knowledge of the local data. Thus with the validation of these experiments, the modules of the proposed method are sufficiently motivated. The visualization results of the average distance of the local models when trained independently with or without the addition of background data are displayed in Figure 5. It is clear from the figures that with the assistance of background data, the independently trained local model is able to obtain a smaller model average distance, effectively slowing down the drift among clients. The experimental results reflect the fact that the background data has a local pre-calibration effect on the federated model at each client, which is a key reason for the effectiveness of the proposed method.

BG	Mix	GG	Twitch	Facebook100	Facebook100-lite
×	×	×	60.55	54.66	52.71

Introducing background data for performance improvement

BG	Mix	GG	Twitch	Facebook100	Facebook100-lite
√	×	×	65.31	57.51	55.64

Used for — The mixing strategy improve privacy but impair performance

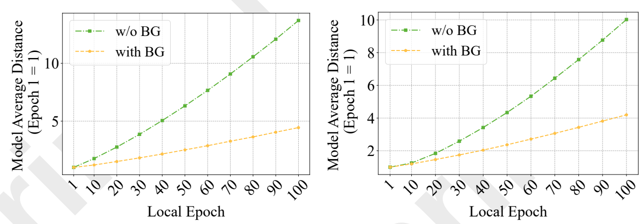
BG	Mix	GG	Twitch	Facebook100	Facebook100-lite
√	√	×	65.15	56.33	54.98

Used for — The diversity of background data is restored by graph generators

BG	Mix	GG	Twitch	Facebook100	Facebook100-lite
√	√	√	65.69	58.20	57.19

	Twitch	Facebook100	Facebook100-lite
Training with BG only	53.07	50.70	50.60

Figure 4: The ablation study with three components. BG: background data, Mix: mixing strategy, GG: graph generation strategy. Mix and GG need to be dependent on BG and Mix, respectively, and are therefore added in sequence so as to explore the sufficiency of the motivation.



(a) Twitch (b) Facebook100-lite

Figure 5: Curves of the average distance among all client models with increasing local epoch in two settings during local training. The use of background data effectively mitigates drift among models.

5 Conclusion

In this paper, we propose a FGL method that uses background graph data to mitigate the bias caused by cross-domain data. The method provides a workable solution to the skew problem in FGL from the data perspective, which improves the effectiveness of cross-domain bias correction from the source rather than from the subsequent part. Background data, as a type of graph data, is naturally capable of combining feature and topological information. The combination of mixing strategy and graph diversity recovery strategy providing additional privacy protection while being able to mine the knowledge of the background data as much as possible, further enhances the utility of the background graph. Experiments are conducted to demonstrate the superiority of the proposed FedBG, as well as the sufficient motivation and effectiveness of each proposed module.

Acknowledgments

The research is supported bythe National Key R&D Program of China (2023YFB2703700), the National Natural Science Foundation of China (62176269).

Contribution Statement

Sheng Huang and Lele Fu contributed equally to this work.

References

[Baek *et al.*, 2023] Jinheon Baek, Wonyong Jeong, Jiongdao Jin, Jaehong Yoon, and Sung Ju Hwang. Personalized subgraph federated learning. In *International conference on machine learning*, pages 1396–1415. PMLR, 2023.

[Cai *et al.*, 2024a] Jinyu Cai, Yunhe Zhan, Zhoumin Lu, Wenzhong Guo, and See-Kiong Ng. Towards effective federated graph anomaly detection via self-boosted knowledge distillation. In *Proceedings of the ACM International Conference on Multimedia*, pages 5537–5546, 2024.

[Cai *et al.*, 2024b] Jinyu Cai, Yunhe Zhang, Jicong Fan, and See-Kiong Ng. Lg-fgad: An effective federated graph anomaly detection framework. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 3760–3769, 2024.

[Fu and King, 2023] Xinyu Fu and Irwin King. Fedhgn: a federated framework for heterogeneous graph neural networks. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 3705–3713, 2023.

[Fu *et al.*, 2025] Lele Fu, Sheng Huang, Yanyi Lai, Tianchi Liao, Chuanfu Zhang, and Chuan Chen. Beyond federated prototype learning: Learnable semantic anchors with hyperspherical contrast for domain-skewed data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 16648–16656, 2025.

[Guo *et al.*, 2023] Jiayan Guo, Shangyang Li, and Yan Zhang. An information theoretic perspective for heterogeneous subgraph federated learning. In *International Conference on Database Systems for Advanced Applications*, pages 745–760. Springer, 2023.

[Hu *et al.*, 2021] Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. Ogb-lsc: A large-scale challenge for machine learning on graphs. *NeurIPS*, 34, 2021.

[Hu *et al.*, 2024] Ming Hu, Yue Cao, Anran Li, Zhiming Li, Chengwei Liu, Tianlin Li, Mingsong Chen, and Yang Liu. Fedmut: Generalized federated learning via stochastic mutation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 12528–12537, 2024.

[Huang *et al.*, 2023a] Wenke Huang, Guancheng Wan, Mang Ye, and Bo Du. Federated graph semantic and structural learning. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 3830–3838, 2023.

[Huang *et al.*, 2023b] Wenke Huang, Mang Ye, Zekun Shi, He Li, and Bo Du. Rethinking federated learning with domain shift: A prototype view. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16312–16322. IEEE, 2023.

[Huang *et al.*, 2024] Sheng Huang, Lele Fu, Yuecheng Li, Chuan Chen, Zibin Zheng, and Hong-Ning Dai. A cross-client coordinator in federated learning framework for conquering heterogeneity. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15, 2024.

[Karimireddy *et al.*, 2020] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *Proceedings of the International Conference on Machine Learning*, pages 5132–5143, 2020.

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

[Li *et al.*, 2021] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021.

[Li *et al.*, 2024] Xunkai Li, Zhengyu Wu, Wentao Zhang, Yinlin Zhu, Rong-Hua Li, and Guoren Wang. Fedgta: Topology-aware averaging for federated graph learning. *arXiv preprint arXiv:2401.11755*, 2024.

[Liao *et al.*, 2022] Jie Liao, Wei Zhou, Fengji Luo, Junhao Wen, Min Gao, Xiuhua Li, and Jun Zeng. Sociallgn: Light graph convolution network for social recommendation. *Information Sciences*, 589:595–607, 2022.

[Liao *et al.*, 2024] Tianchi Liao, Lele Fu, Jialong Chen, Zhen Wang, Zibin Zheng, and Chuan Chen. A swiss army knife for heterogeneous federated learning: Flexible coupling via trace norm. In *Advances in Neural Information Processing Systems*, volume 37, pages 139886–139911, 2024.

[Lin *et al.*, 2020] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Proceedings of the Conference and Workshop on Neural Information Processing Systems*, pages 2351–2363, 2020.

[Liu *et al.*, 2023] Chengliang Liu, Zhihao Wu, Jie Wen, Yong Xu, and Chao Huang. Localized sparse incomplete multi-view clustering. *IEEE Transactions on Multimedia*, 25:5539–5551, 2023.

[Liu *et al.*, 2025] Chengliang Liu, Jie Wen, Yong Xu, Bob Zhang, Liqiang Nie, and Min Zhang. Reliable representation learning for incomplete multi-view missing multi-label classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–17, 2025.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 1273–1282, 2017.

[Peng *et al.*, 2022] Liang Peng, Nan Wang, Nicha Dvornek, Xiaofeng Zhu, and Xiaoxiao Li. Fedni: Federated graph learning with network inpainting for population-based disease prediction. *IEEE Transactions on Medical Imaging*, 42(7):2032–2043, 2022.

[Qi *et al.*, 2023] Zhuang Qi, Lei Meng, Zitan Chen, Han Hu, Hui Lin, and Xiangxu Meng. Cross-silo prototypical calibration for federated learning with non-iid data. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3099–3107, 2023.

[Qi *et al.*, 2025] Zhuang Qi, Lei Meng, and et al. Cross-silo feature space alignment for federated learning on clients with imbalanced data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 19986–19994, 2025.

[Wan *et al.*, 2024] Guancheng Wan, Wenke Huang, and Mang Ye. Federated graph learning under domain shift with generalizable prototypes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 15429–15437, 2024.

[Wang *et al.*, 2020] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. In *Proceedings of the Conference and Workshop on Neural Information Processing Systems*, pages 7611–7623, 2020.

[Wang *et al.*, 2022a] Binghui Wang, Ang Li, Meng Pang, Hai Li, and Yiran Chen. Graphfl: A federated learning framework for semi-supervised node classification on graphs. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 498–507. IEEE, 2022.

[Wang *et al.*, 2022b] Zhen Wang, Weirui Kuang, Yuexiang Xie, Liuyi Yao, Yaliang Li, Bolin Ding, and Jingren Zhou. Federatedscope-gnn: Towards a unified, comprehensive and efficient package for federated graph learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4110–4120, 2022.

[Wu *et al.*, 2022] Qitian Wu, Hengrui Zhang, Junchi Yan, and David Wipf. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466*, 2022.

[Xia *et al.*, 2022] Wei Xia, Quanxue Gao, Qianqian Wang, Xinbo Gao, Chris Ding, and Dacheng Tao. Tensorized bipartite graph learning for multi-view clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):5187–5202, 2022.

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

[Xiong *et al.*, 2023] Yuanhao Xiong, Ruochen Wang, Minhao Cheng, Felix Yu, and Cho-Jui Hsieh. Feddm: Iterative distribution matching for communication-efficient federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16323–16332, 2023.

[Yang *et al.*, 2021] Shuo Yang, Zhiqiang Zhang, Jun Zhou, Yang Wang, Wang Sun, Xingyu Zhong, Yanming Fang, Quan Yu, and Yuan Qi. Financial risk analysis for smes with graph-based supply chain mining. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 4661–4667, 2021.

[Yang *et al.*, 2022] Haizhou Yang, Quanxue Gao, Wei Xia, Ming Yang, and Xinbo Gao. Multiview spectral clustering with bipartite graph. *IEEE Transactions on Image Processing*, 31:3591–3605, 2022.

[Ye *et al.*, 2023] Mang Ye, Xiuwen Fang, Bo Du, Pong C Yuen, and Dacheng Tao. Heterogeneous federated learning: State-of-the-art and research challenges. *ACM Computing Surveys*, 56(3):1–44, 2023.

[Yi *et al.*, 2022] Hai-Cheng Yi, Zhu-Hong You, De-Shuang Huang, and Chee Keong Kwoh. Graph representation learning in bioinformatics: trends, methods and applications. *Briefings in Bioinformatics*, 23(1):bbab340, 2022.

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

[Zhang *et al.*, 2024a] Chen Zhang, Yu Xie, Tingbin Chen, Wenjie Mao, and Bin Yu. Prototype similarity distillation for communication-efficient federated unsupervised representation learning. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[Zhang *et al.*, 2024b] Jianqing Zhang, Yang Liu, Yang Hua, and Jian Cao. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16768–16776, 2024.

[Zhang *et al.*, 2024c] Ke Zhang, Lichao Sun, Bolin Ding, Siu Ming Yiu, and Carl Yang. Deep efficient private neighbor generation for subgraph federated learning. In *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*, pages 806–814. SIAM, 2024.

[Zhu *et al.*, 2024] Yinlin Zhu, Xunkai Li, Zhengyu Wu, Di Wu, Miao Hu, and Rong-Hua Li. Fedtad: Topology-aware data-free knowledge distillation for subgraph federated learning. *arXiv preprint arXiv:2404.14061*, 2024.

[Zhuang *et al.*, 2024] Shuman Zhuang, Sujia Huang, Wei Huang, Yuhong Chen, Zhihao Wu, and Ximeng Liu. Enhancing multi-view graph neural network with cross-view confluent message passing. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 10065–10074, 2024.