MULTIKD: Backdoor Defense in Federated Graph Learning via Attention-Guided Multi-Teacher Distillation

Jiale Zhang[1*], Yanan Wang[1], Bosen Rao[1], Chengcheng Zhu[2], Xiaobing Sun[1], Yu Li[3]

[1] School of Information and Artificial Intelligence, Yangzhou University, China
[2] State Key Laboratory for Novel Software Technology, Nanjing University, China
[3] School of Artificial Intelligence, Jilin University, China
{jialezhang, xbsun}@yzu.edu.cn, {MX120240570, MX120230573}@stu.yzu.edu.cn, chengchengzhu@smail.nju.edu.cn, liyu90@jlu.edu.cn

Abstract

Backdoor attacks pose a severe threat to federated graph learning (FGL), where malicious clients can inject hidden triggers into the global model without being detected. Defending against such attacks is particularly challenging due to the complex graph structures and the stealthy nature of trigger patterns. In this work, we propose MULTIKD, a novel backdoor mitigation method based on attention-guided multi-teacher distillation. Unlike existing defenses that focus on detecting suspicious clients or restricting backdoor activation, MULTIKD directly purifies the global model on the server side by exploiting intermediate representations. It integrates knowledge from multiple client models and guides the global model to suppress backdoor behaviors by aligning attention maps and preserving inter-layer relational consistency. Our defensive intuition enables MULTIKD to retain task-relevant information while mitigating malicious patterns, even when some teacher models are compromised. Extensive experiments on four real-world datasets demonstrate the effectiveness of our approach in significantly reducing attack success rate ($\leq 8\%$) with minimal impact on utility ($\leq 5\%$).

Introduction

Federated Learning (FL) (Yang et al. 2019; Mammen 2021) has emerged as a promising paradigm for collaborative training across decentralized clients without sharing raw data, effectively addressing data privacy and silo issues (Li, Sharma, and Mohanty 2020). With the growing need to process massive graph-structured data in many fields, including social networks, recommendation systems, and financial systems (Huang et al. 2022; Liu et al. 2024), Federated Graph Learning (FGL) extends FL into the graph domain by integrating Graph Neural Networks (GNNs) (Ju et al. 2024). FGL enables clients to perform local GNN training and share model parameters with a central server, thereby supporting efficient learning across distributed graph data. However, the reliance on model updates provided by clients also exposes FGL to security threats, particularly backdoor attacks from malicious participants (Chen et al. 2017; Li et al. 2022).

In FGL, attackers usually disguise themselves as normal clients and inject specific triggers into their local training data, tampering with the corresponding labels. As a result, the backdoor will be implicitly integrated into the global parameters during aggregation, allowing the global model to behave normally on benign inputs while producing attacker-specified predictions when encountering the trigger. For example, centralized backdoor attack (CBA) and distributed backdoor attack (DBA) are two representative strategies (Xu et al. 2022). CBA embeds a global trigger into one malicious client's data, while DBA decomposes the trigger into local patterns distributed across multiple clients. Another line of work introduces adaptive subgraph-based triggers that dynamically optimize their position and shape to improve stealth and effectiveness (Yang et al. 2024).

Some defense methods have been proposed in FL, such as FLAME (Nguyen et al. 2022) and FoolsGold (Fung, Yoon, and Beschastnikh 2020), which aim to detect suspicious clients by measuring update similarity or behavioral consistency among participants. However, these approaches are difficult to apply effectively in FGL. Due to the complex graph structure and the stealthiness and diversity of triggers, malicious behaviors become difficult to detect and distinguish. To address these limitations, recent works have proposed defense strategies specifically for FGL. One study (Yang et al. 2024) introduces a certified defense by splitting the test graph into subgraphs and applying majority voting to restrict backdoor activation in the testing phase. Another recent work (FedTGE) (Wan et al. 2025) introduces an energy-based clustering and propagation framework to identify and isolate malicious clients based on their energy distribution patterns. While both approaches consider graph-specific characteristics, one remains confined to the inference phase, and the other centers on detecting and filtering anomalous clients. However, they tend to overlook the global model itself. Once backdoor information enters or stays in the model, its influence becomes difficult to remove.

Motivation: In light of the above limitations, we are motivated by two key observations: ❶ Most FL defense methods perform poorly in FGL, as complex graph structures and diverse triggers often obscure backdoor patterns. Specifically, in graph models, triggers can subtly alter structural and feature information across layers, making them difficult to detect through straightforward behavioral analysis. However, many existing approaches overlook the propagation

*Corresponding author.

and transformation of such information within intermediate representations, which is critical to understanding and mitigating backdoor effects in graph learning. ❷ Recent FGL defenses tend to focus on preventing the implantation or activation of backdoors. However, effective defense should not be limited to evasion alone, but should also involve direct intervention, particularly when the global model has already been compromised. Therefore, a more practical approach is to directly purify the global model within the FGL process.

Challenges: In summary, there are two key challenges to design an effective backdoor mitigation strategy tailored for FGL: ❶ how to effectively suppress backdoor effects in the global model under a multi-client setting; ❷ how to guide the global model to form more robust intermediate representations and weaken the persistence and propagation of backdoors within the model.

In this work, we propose MULTIKD, a server-side defense method customized for FGL. MULTIKD adopts an attention-guided multi-teacher distillation strategy: all uploaded clients' models are treated as teacher models, and their knowledge is distilled into the global model, which acts as the student model. With the aid of a small set of clean data, this process purifies the global model and effectively suppresses injected backdoors. To extract informative guidance from teacher models, MULTIKD distills not only output predictions but also intermediate representations. Specifically, Attention Map Alignment encourages the student to match the attention distributions of teachers, reinforcing attention consistency and mitigating backdoor-induced attention shifts. In parallel, Inter-layer Relation Consistency captures the structural dependencies across layers, guiding the student to maintain stable and robust feature propagation. Together, these techniques lead to cleaner intermediate representations and significantly reduce residual backdoor effects in the global model. The key contributions include:

- **A novel backdoor defense method:** We propose MULTIKD, a server-side backdoor mitigation method for FGL, which suppresses backdoor threats by directly purifying the global model.
- **Multi-teacher knowledge distillation:** To accommodate the multi-client setting in FGL, MULTIKD leverages a many-to-one distillation strategy, where client models serve as teacher models to jointly guide the global model.
- **Attention-guided representation learning:** We introduce two complementary distillation modules, Attention Map Alignment and Inter-layer Relation Consistency, which align intermediate representations and structural relations to suppress the influence of backdoor triggers.
- **Comprehensive evaluation:** Extensive experiments on four benchmark datasets validate the effectiveness of MULTIKD under various backdoor attacks, outperforming state-of-the-art defenses.

Related Work

Backdoor Attacks in FGL

Backdoor attacks in FL are commonly categorized into centralized and distributed backdoor attacks (CBA and DBA). Centralized backdoor attacks typically employ a shared trigger across malicious clients to overwrite the global model via techniques such as model replacement (Bagdasaryan et al. 2020). In contrast, distributed backdoor attacks assign different local triggers to compromised clients, which collectively inject a global backdoor into the model (Xie et al. 2019). These attack paradigms have been further adapted to FGL, where the graph-structured data introduces new challenges and vulnerabilities. A representative study extends CBA and DBA to the graph domain, demonstrating that both attack types remain effective under FGL settings (Xu et al. 2022). Another work employs adaptive trigger generators that optimize trigger placement and shape based on graph features for more concealed attacks (Yang et al. 2024). Moreover, a recent approach proposes a non-intrusive attack that trains a perturbation generator on clean data, leveraging natural vulnerabilities in the global model without modifying local training data (Li et al. 2025). The continued advancement of backdoor attacks in FGL underscores the need for robust and generalizable defense mechanisms.

Backdoor Defenses in FGL

In FL, one class of defenses attempts to detect and filter abnormal client behaviors based on discrepancies in model updates, gradients, or other metrics, such as Krum (Blanchard et al. 2017) and FoolsGold (Fung, Yoon, and Beschastnikh 2020). However, these passive methods can only help identify potential backdoor threats, while struggling to effectively remove their negative impact on the global model. Some defenses attempt to mitigate backdoors by suppressing or perturbing local updates, as done in FLAME (Nguyen et al. 2022), RLR (Ozdayi, Kantarcioglu, and Gel 2021), and similar approaches (Xie et al. 2021). However, the properties of graph data, including irregular structures, inconsistent feature distributions, and diverse semantic patterns, often compromise their effectiveness in FGL settings. Consequently, in FGL, one certified method (CertGNN) adopts a subgraph voting strategy during inference to suppress backdoor activation (Yang et al. 2024), while another (FedTGE) detects and filters anomalous clients based on energy distribution patterns (Wan et al. 2025). Our method, by contrast, directly purifies the global model via multi-teacher distillation, mitigating backdoor threats that remain embedded during aggregation, which prior defenses inevitably face.

Background and Threat Model

Federated Graph Learning

A graph is represented as $G = (V, E, X)$, where $V = \{v_1, v_2, ..., v_N\}$ is the set of nodes, E is the set of edges, and $X \in \mathbb{R}^{N \times d}$ denotes the node feature matrix with d-dimensional features for each of the N nodes. The graph structure is encoded by an adjacency matrix $A \in \mathbb{R}^{N \times N}$, where $A_{ij} = 1$ indicates that an edge exists between nodes v_i and v_j, and $A_{ij} = 0$ otherwise. In graph classification tasks, each graph G is associated with a label $y \in Y$, where Y is the set of class labels. The objective of graph learning is to train a classifier f that maps an input graph to its corresponding label, i.e., $f(G) \to y$.

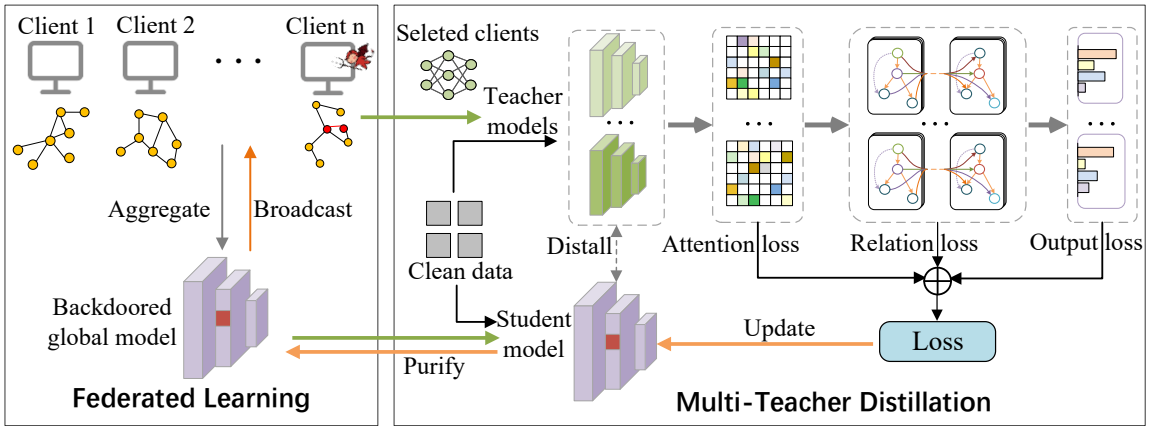


Figure 1: Framework of the proposed MULTIKD.

FGL adapts the FL paradigm to graph-structured data, where a central server coordinates the training of a global GNN model F across multiple clients without sharing local graph data. Given a collection of clients $\mathcal{C} = \{1, 2, \ldots, M\}$, each client $i \in \mathcal{C}$ has a local graph dataset $\mathcal{D}_i = \{(G_1^i, y_1^i), \ldots, (G_{n_i}^i, y_{n_i}^i)\}$. In a t-th round, the server selects a subset of clients $S_t \subseteq \mathcal{C}$ and broadcasts the current global model F_t to them. Each client $i \in S_t$ updates its model using local training data $\mathcal{D}_i$, based on the received global model. The update follows a standard gradient descent step: $F_t^i \leftarrow F_t - \eta \nabla L(\mathcal{D}_i; F_t)$, where L is the local loss function (e.g., cross-entropy loss) and η is the learning rate. After receiving updated models from all participating clients, the server aggregates them to compute the global model F_{t+1} for the next iteration. A common aggregation strategy is Federated Averaging (FedAvg), defined as $F_{t+1} = \frac{1}{|S_t|} \sum_{i \in S_t} F_t^i$, where $|S_t|$ denotes the number of selected clients. This process repeats iteratively until the global model converges or the training reaches the communication round limit.

Threat Model

We focus on a malicious scenario in FGL, where a portion of participants behave as internal adversaries and attempt to inject backdoor behavior into the global model by uploading carefully manipulated local updates. We define the attacker's capability and goals as follows:

Attacker's capability. Malicious clients have control over the local training data and procedures. However, they cannot access any information from benign clients and interfere with the aggregation process conducted by the server.

Attacker's goals. The malicious clients aim to steer the global model toward backdoor behavior. Consequently, the aggregated model maintains high performance on clean inputs while producing attacker-specified outputs when triggered by predefined patterns.

Similar to previous backdoor defense studies (Fung, Yoon, and Beschastnikh 2020; Lu et al. 2022), we define the defender's capability and goals as follows:

Defender's capability. The defender receives model updates from clients and has no access to local data or any knowledge about backdoor behavior. In addition, the defender holds a small portion of labeled test data on the server side, which serves as clean data for the defense process.

Defender's goals. The defense goal is to effectively suppress backdoor in the global model, significantly reducing the attack success rate (ASR) on triggered inputs while preserving the clean accuracy (ACC) on benign samples.

Design of MULTIKD

Overview

In this section, we present MULTIKD, a novel defense strategy against backdoor attacks in FGL. The overall workflow of MULTIKD is illustrated in Fig. 1. The core idea is to leverage a small amount of clean data on the server to perform multi-teacher knowledge distillation, thereby purifying the global model aggregated from potentially malicious client updates. At each communication round, the server treats all selected client models as teacher models and uses the aggregated global model as the student model. Although some teachers may be malicious, we rely on clean inputs during the distillation process. Given that backdoor models generally behave normally on benign samples, their internal representations (e.g., attention maps and inter-layer dependencies) can still reflect valid task-relevant knowledge. This allows the student model to distill generalized patterns even without explicit teacher filtering.

During the distillation process, Attention Map Alignment encourages the student model to mimic the attention distributions of teacher models, which highlight influential nodes and their roles in message passing, suppressing potential attention deviations introduced by backdoor triggers. Inter-layer Relation Consistency transfers structural dependencies between layers, helping the student model maintain stable feature propagation. In addition, Label Supervision anchors the training to the original task objective, ensuring predictive performance is preserved.

Attention Map Alignment

To align the intermediate representations between the student and teachers, we performed attention-based distillation across multiple layers. In GNNs, attention maps highlight which nodes are more informative during message passing, typically reflecting task-relevant semantics in benign models. However, poisoned models often exhibit abnormal attention concentration induced by backdoor triggers. By aligning the attention distributions obtained on clean inputs, the student is guided to suppress spurious focus introduced by malicious clients and recover reliable attention behaviors.

Given a selected client $i \in S_t$, we treat its uploaded model F_i as a teacher model and the aggregated global model F as the student model. We denote the activation tensor in the l-th layer of the teacher model as $H_i^l \in \mathbb{R}^{N \times C_l}$, where N represents the number of nodes and C_l is the channel dimension. To obtain attention maps, we define an attention mapping function $\mathcal{A} : \mathbb{R}^{N \times C_l} \to \mathbb{R}^N$ that transforms each activation tensor into an attention representation. Formally,

$$\mathcal{A}(H_i^l) = \sum_{j=1}^{C_l} \left| H_i^{l(j)} \right|^2, \tag{1}$$

where $H_i^{l(j)}$ denotes the activation map of the j-th channel, $|\cdot|$ is the absolute value function.

To ensure scale consistency across different layers and inputs, we normalize the attention representation. This step allows the distillation to focus on distributional patterns rather than absolute magnitudes, enabling more stable and comparable alignment. We apply the normalization as follows:

$$\hat{\mathbf{A}}_i^{\,l} = \frac{\mathcal{A}(H_i^l) - \mu_i^l}{\sigma_i^l + \epsilon}, \tag{2}$$

where μ_i^l and σ_i^l denote the mean and standard deviation of $\mathcal{A}(H_i^l)$, and ϵ is a small constant to avoid division by zero.

Then, we compute the alignment loss between the student and each teacher model. To account for possible variations across teacher models, we compute pairwise attention alignment losses and average the results. Formally,

$$\mathcal{L}_{\text{att}}^{(l)} = \frac{1}{|S_t|} \sum_{i \in S_t} \left\| \hat{\mathbf{A}}_{\text{student}}^{\,l} - \hat{\mathbf{A}}_i^{\,l} \right\|_2^2, \tag{3}$$

where $\hat{\mathbf{A}}_{\text{student}}^{\,l}$ and $\hat{\mathbf{A}}_i^{\,l}$ denote the normalized attention maps of the student model and the i-th teacher model at layer l. $|S_t|$ indicates the number of participating teacher models. The total attention alignment loss is computed as follows:

$$\mathcal{L}_{\text{att}} = \sum_{l=1}^{L} \lambda_l \cdot \mathcal{L}_{\text{att}}^{(l)}, \tag{4}$$

where λ_l denotes the weight assigned to the l-th layer. The use of layer-wise weights allows the distillation process to emphasize more informative layers, contributing to improved generalization and robustness.

Inter-layer Relation Consistency

We introduce attention map alignment to help the student model focus on task-relevant regions and defend against potential backdoors. However, relying only on attention may be insufficient under the constraint of limited clean data, where the student model may fail to fully capture the structural knowledge encoded in the teacher models. To mitigate this, we guide the student model to align with the relational structure formed between layers in teacher models. By maintaining relation consistency, the student can better preserve the feature transformation pattern of benign models.

Specifically, given the normalized attention maps $\hat{\mathbf{A}}^1, \hat{\mathbf{A}}^2, \ldots, \hat{\mathbf{A}}^L$ from a model, we compute the cosine similarity between all pairs of layers to obtain a relation matrix $R \in \mathbb{R}^{L \times L}$. Each element R_{ij} measures the similarity between layer i and layer j:

$$R_{ij} = \cos\left(\hat{\mathbf{A}}^i, \hat{\mathbf{A}}^j\right) = \frac{\hat{\mathbf{A}}^i \cdot \hat{\mathbf{A}}^j}{\|\hat{\mathbf{A}}^i\|_2 \cdot \|\hat{\mathbf{A}}^j\|_2}. \tag{5}$$

Let R^S and R_i^T denote the relation matrices computed from the student model and the i-th teacher model, respectively. The relation loss is defined as

$$\mathcal{L}_{\text{rel}} = \frac{1}{|S_t|} \sum_{i \in S_t} \left\| R^S - R_i^T \right\|_2^2. \tag{6}$$

Label Supervision

We additionally use clean data with its true labels to guide the student model's predictions. While insufficient on its own to eliminate backdoors, this objective reinforces correct classification and helps stabilize the distillation process. This objective can be formulated as

$$\mathcal{L}_{\text{sup}} = \mathcal{L}_{\text{CE}}(F_S(x),\ y), \tag{7}$$

where $\mathcal{L}_{\text{CE}}$ denotes the cross-entropy loss, $F_S(x)$ is the prediction of the student model. The final optimization objective of MULTIKD is to minimize the following total loss:

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \alpha \cdot \mathcal{L}_{\text{att}} + \beta \cdot \mathcal{L}_{\text{rel}}, \tag{8}$$

where α and β are hyperparameters that balance the contribution of attention alignment and relation consistency.

After the distillation process, the student model is adopted as the new global model and broadcast to clients for the next round. In this way, MULTIKD progressively eliminates backdoors while reinforcing robust feature representations.

Evaluation

In this section, we evaluate the effectiveness of MULTIKD on the federated graph classification task under various backdoor attack scenarios.

Experimental Setup

Datasets. We evaluate our approach on four real-world datasets: PROTEINS (Borgwardt et al. 2005), NCI1 (Shervashidze et al. 2011), DD (Shervashidze et al. 2011), and AIDS (Rossi and Ahmed 2015). PROTEINS and DD are

Datasets	Attacks	NoDefense		FLAME		FoolsGold		CertGNN		FedTGE		MULTIKD	
		ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR	ACC	ASR
PROTEINS	**Subgraph**	67.11	24.09	66.64	24.63	66.53	28.63	61.88	6.94	**73.28**	30.26	70.15	**4.30**
	UGBA	62.27	73.88	75.12	8.16	**76.62**	13.27	75.61	14.56	73.66	72.91	72.14	**0.12**
	Motif	64.73	78.12	**74.13**	9.38	73.13	14.58	60.68	28.61	72.27	42.07	69.65	**2.08**
	Opt-GDBA	72.86	82.33	**73.16**	12.33	72.69	15.67	62.88	35.63	72.52	34.70	69.39	**0.54**
NCI1	**Subgraph**	61.92	31.68	**72.96**	22.58	70.06	21.51	67.43	21.33	68.75	35.63	69.73	**8.60**
	UGBA	64.16	89.12	72.48	14.29	**72.95**	11.22	65.97	17.65	64.79	17.68	69.02	**3.06**
	Motif	61.71	67.93	**72.86**	11.46	**72.86**	11.46	66.72	34.21	63.67	36.39	69.96	**6.25**
	Opt-GDBA	70.42	80.06	69.11	58.45	**72.41**	86.15	68.21	73.99	70.44	23.71	69.39	**0.11**
DD	**Subgraph**	67.12	30.54	71.43	8.33	**77.78**	15.48	62.49	8.47	67.84	10.25	75.13	**7.14**
	UGBA	69.21	98.86	74.07	14.77	**76.19**	12.50	57.24	73.06	64.96	81.22	75.66	**7.95**
	Motif	64.97	64.63	74.60	13.33	**77.25**	21.11	62.24	27.08	64.26	21.70	76.19	**7.78**
	Opt-GDBA	71.34	78.53	70.81	77.64	70.64	75.63	60.44	61.98	65.53	72.64	**75.14**	**6.84**
AIDS	**Subgraph**	99.56	24.56	**99.76**	4.16	**99.76**	1.56	95.25	14.06	91.96	1.88	95.83	**0.54**
	UGBA	99.44	89.50	**99.44**	94.54	**99.44**	93.99	96.63	94.18	93.41	64.63	**99.44**	**3.57**
	Motif	99.88	38.45	91.11	10.99	**98.07**	4.05	92.26	8.90	93.97	1.79	97.50	**1.32**
	Opt-GDBA	99.11	44.62	98.21	43.24	98.05	50.34	95.06	37.21	91.85	17.03	**99.51**	**1.12**

Table 1: Performance of MULTIKD compared with baseline methods on four datasets.

bioinformatics datasets where each graph represents a protein structure, with nodes denoting secondary structure elements and edges indicating spatial or functional proximity. NCI1 is a chemical compound dataset for binary classification of anti-cancer activity. AIDS is a molecular graph dataset where nodes represent atoms and edges denote chemical bonds, with the task of classifying compounds based on their effectiveness against the AIDS virus.

Comparison Methods. We compare MULTIKD with four representative defense methods: FLAME (Nguyen et al. 2022), FoolsGold (Fung, Yoon, and Beschastnikh 2020), CertGNN (Yang et al. 2024), and FedTGE (Wan et al. 2025). Among them, FLAME and FoolsGold are designed for FL, while CertGNN and FedTGE are tailored for FGL.

Attack Methods. We evaluate our defense against four types of backdoor attacks: Subgraph (Xu et al. 2022), UGBA (Dai et al. 2023), Motif (Zheng et al. 2023), and Opt-GDBA (Yang et al. 2024). Subgraph and Motif attacks generate structural triggers by injecting specific subgraph patterns, where Motif is based on motif frequency and injection strategy. UGBA introduces adaptive and stealthy triggers that are harder to detect. Opt-GDBA assigns each malicious client a locally optimized trigger, representing a strong and dynamic attack strategy in federated settings.

Models. To evaluate the defensive effectiveness of MULTIKD, we adopt three widely used GNN architectures: Graph Convolutional Network (GCN) (Kipf 2017), Graph Isomorphism Network (GIN) (Xu et al. 2019), and Graph Attention Network (GAT) (Veličković et al. 2018). Note that GCN is the default model unless otherwise specified.

Evaluation Metrics. We adopt two standard metrics to evaluate defense capability: ASR and ACC. ASR measures the proportion of poisoned inputs that are misclassified into the target label (the lower the better), while ACC indicates the model's prediction accuracy on clean test data (the higher the better).

Implementation Details. All experiments are implemented MULTIKD in Python using the PyTorch framework. The experimental environment consists of 13th Gen Intel(R) Core(TM) i7-13700KF, NVIDIA GeForce RTX 4070 Ti, 32GiB memory, and Ubuntu 20.04 (OS). The data is split into a training set and a test set with an 80:20 ratio, wherein the training set is evenly distributed among all participants in the FGL framework. We assume that MULTIKD has random access to 5% of the clean data in the test set. We set 10 rounds of local optimization for FL with a batch size of $B = 128$ and a learning rate of $\eta = 0.01$ for local training in FL. We define 10 participants, from which 5 are randomly selected for aggregation in each round. For the distillation process, the loss terms are set to $\alpha = \beta = 0.5$, Adam is used as the optimizer with a learning rate of $\eta = 0.001$, and it is run for $E = 10$ epochs. For all baseline attack and defense methods, we adopt the default hyperparameters recommended in their corresponding papers. Specifically, the attacks share common parameters: trigger size t and injection rate φ. Given a training set D_{train} with an average node count of N_{avg}, the number of nodes in the subgraph trigger is equal to $N_{avg} \times t$. Unless otherwise specified, the backdoor injection rate is set to $\varphi = 50\%$, and the trigger size is set to $t = 30\%$. We test the performance of MULTIKD as well as other baseline methods five times and report the average values to eliminate the influence of randomness.

Experimental Results

Comparison. Table 1 presents the performance of MULTIKD and four baseline defense methods across four datasets. The column "NoDefense" is listed as the original baseline without any defense, and the best results are

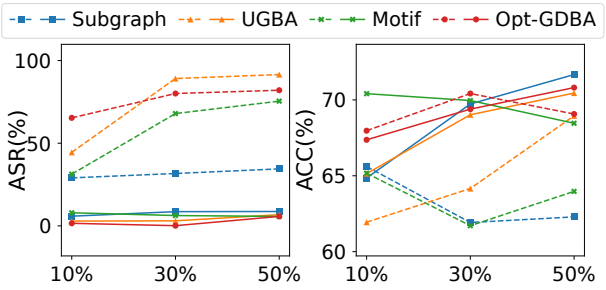


Figure 2: Impact of trigger ratio on NCI1 dataset.

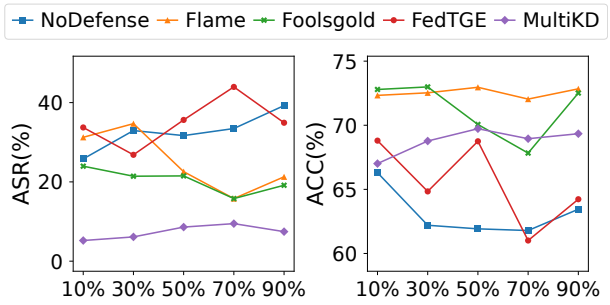


Figure 3: Impact of inject ratio on Subgraph attack.

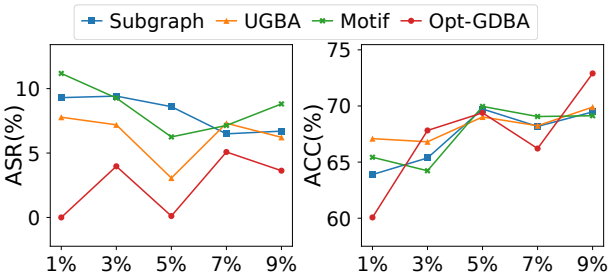


Figure 4: Impact of clean data ratio on NCI1 dataset.

presented in bold. Overall, MULTIKD consistently achieves low ASR across all datasets and attack types, particularly on the AIDS and PROTEINS datasets where ASR drops to nearly zero. This indicates its strong capability in neutralizing backdoor effects. In addition, MULTIKD achieves comparable or even higher ACC than NoDefense, reflecting a more effective integration of benign knowledge from diverse client models. Among the baseline methods, FLAME and FoolsGold demonstrate similar performance, with relatively high ACC across various settings. This is likely due to their ability to identify and reduce the influence of suspicious clients, thereby preserving benign updates. However, their ASR is less stable, remaining high under UGBA and Opt-GDBA attacks. This suggests that such defenses may struggle against more adaptive or stealthy poisoning strategies that bypass simple client-level screening. Moreover, CertGNN yields comparatively lower ACC and slightly higher ASR than other baselines. We suspect that its subgraph partitioning and majority voting mechanism may interfere with structural semantics and lead to inconsistent predictions between subgraphs, thereby affecting classification accuracy. The elevated ASR further indicates that CertGNN applied at the inference stage may be insufficient to eliminate backdoor effects. FedTGE, though designed for FGL scenarios, shows uneven performance. It improves ACC on PROTEINS and NCI1 but underperforms on DD and AIDS. Its ASR remains high, particularly under the UGBA attack and on the PROTEINS dataset. This reflects potential challenges in distinguishing malicious clients under complex topologies or subtle trigger semantics, which can confound energy-based discrimination. In summary, MULTIKD maintains a favorable balance between ASR and ACC, and outperforms other defenses. It effectively suppresses backdoor effects while preserving clean utility.

Impact of Trigger Ratio. We further evaluated the impact of trigger ratio on MULTIKD across four attack types. Notably, for the Motif attack, we select different motifs (M41, M43, and M45) instead of trigger ratio. The results on NCI1 are shown in Fig. 2, where solid and dashed lines denote performance with and without our defense. In the absence of defense, ASR increases with larger trigger ratios, which is expected since larger triggers introduce stronger poisoning signals. In contrast, MULTIKD consistently suppresses ASR to a low and stable level, demonstrating strong resilience against varying trigger sizes. As for ACC, our method maintains competitive performance under different attack types, affirming the utility of MULTIKD.

Impact of Inject Ratio. We investigated the robustness of MULTIKD under varying backdoor injection ratios ranging from 10% to 90% against the Subgraph attack on the NCI1 dataset. As shown in Fig. 3, MULTIKD generally maintains low ASR across different injection levels, outperforming other baselines. Meanwhile, its ACC fluctuates only slightly and remains higher than the NoDefense setting. The robustness of MULTIKD can be attributed to the use of attention-guided multi-teacher distillation, which encourages the student model to align with clean knowledge from multiple sources. As the injection ratio increases, poisoned models exhibit greater inconsistency in attention and relational structures, making it easier for the distillation to identify and suppress malicious patterns.

Impact of Clean Data Ratio. We evaluated how the availability of clean data affects the effectiveness of MULTIKD. Fig. 4 presents the results on the NCI1 dataset. With an increasing clean data ratio, ASR generally shows a slight downward trend, while ACC gradually improves. Overall, MULTIKD demonstrates strong tolerance to limited clean data, making it applicable in practical FGL settings. We argue that the tolerance stems from the role of clean data in the distillation process: rather than relying on a large number of clean samples to cover the full data distribution, MULTIKD leverages them as anchor signals to guide the student model away from suspicious or inconsistent patterns introduced by malicious clients. Once this corrective signal is present, even in small amounts, it can sufficiently bias the global update toward benign behaviors.

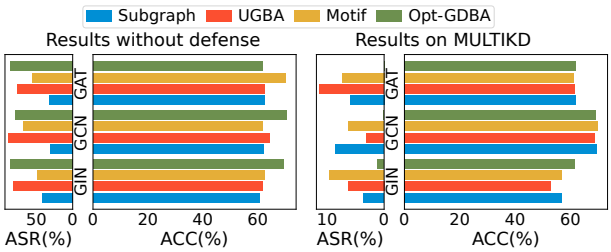


Figure 5: Impact of model types.

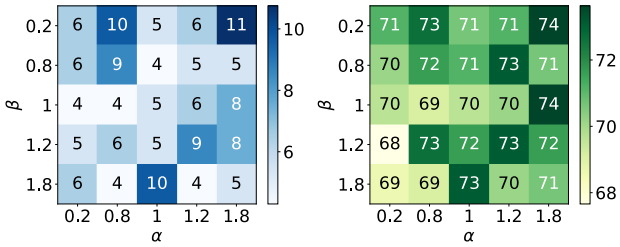
Attacks	Before		FedTGE		MULTIKD	
	ACC	ASR	ACC	ASR	ACC	ASR
Subgraph	63.67	37.39	61.55	48.61	63.22	9.69
UGBA	65.72	66.48	67.49	96.45	59.46	1.03
Motif	63.82	47.19	64.45	21.15	60.10	8.58
Opt-GDBA	64.90	70.85	70.24	32.49	60.04	0.51

Table 2: Performance of MULTIKD with Non-IID settings.

Impact of Model Types. Fig. 5 shows the defense performance on different GNN models under various backdoor attacks. In the absence of defense, all models exhibit clear vulnerabilities to different attacks, but with varying degrees of susceptibility. After applying MULTIKD, we observe a substantial reduction in ASR across all model types. GCN achieves consistently low ASR under all attacks, and GAT obtains near-zero ASR in Opt-GDBA attack. This is likely because our defense explicitly aligns the attention behaviors through attention map distillation, effectively neutralizing the abnormal attention shift induced by backdoor triggers.

Non-IID Data Distributions. To evaluate robustness in practical federated settings, we simulated non-independent and identically distributed (non-IID) data splits and compared defense performance across attacks, as reported in Table 2. MULTIKD achieves lower ASR than other methods, especially under UGBA and Opt-GDBA, showing effectiveness even when client distributions diverge. For ACC, MULTIKD is slightly below FedTGE in some cases. This may be due to the inherent inconsistency among local models in non-IID settings, which introduces variation in the distilled knowledge and affects the generalization of the global model. Nevertheless, MULTIKD remains a more dependable defense under non-IID settings.

Loss Terms. Fig. 6 illustrates how varying the weights α and β, impacts the defense performance of MULTIKD. The results show that MULTIKD achieves the lower ASR and maintains competitive ACC when both weights are set to moderate values, suggesting that a balance between the two weights leads to optimal robustness. In contrast, overemphasizing one loss while underweighting the other causes performance degradation. For example, when α is high and β is low, the ASR increases, indicating that relying too heavily on a single form of knowledge is insufficient to fully suppress the backdoor behavior. The synergy between them enables the student model to better mimic clean knowledge while resisting malicious patterns.



(a) Heatmap of ASR. (b) Heatmap of ACC.

Figure 6: Impact of loss term α and β.

$\mathcal{L}_{\text{att}}$	$\mathcal{L}_{\text{rel}}$	NCI1		PROTEINS		DD	
		ACC	ASR	ACC	ASR	ACC	ASR
✓	✗	60.84	25.62	62.84	42.62	67.22	34.57
✗	✓	63.49	17.70	65.49	14.12	63.14	15.26
✓	✓	62.07	1.85	69.39	0.54	75.14	6.84

Table 3: Performance of MULTIKD with different components.

Component Contributions. We performed an ablation study to investigate the individual contributions of the two components in our framework: Attention Map Alignment and Inter-layer Relation Consistency, as shown in Table 3. When only Inter-layer Relation Consistency is used, the model significantly reduces ASR by preserving the intrinsic relational patterns between layers. In addition, Inter-layer Relation Consistency also leads to higher ACC on NCI1 and PROTEINS, while Attention Map Alignment performs better on DD. Overall, combining both components yields the best performance, achieving the lowest ASR while also improving ACC. These results underscore the complementary strengths of the two modules and validate their joint contribution to robust and generalizable model distillation.

Conclusion & Discussion

In this paper, we present MULTIKD, a defense framework tailored for FGL, which mitigates backdoor threats through attention-guided multi-teacher distillation. By distilling both attention patterns and structural relations from multiple client models into the global model with clean data, MULTIKD encourages more consistent and robust representations that resist the influence of injected backdoors. Experimental results on multiple benchmarks confirm its effectiveness across diverse attack scenarios. Future research directions will focus on optimizing the computational efficiency of the distillation process and reducing the reliance on clean data, further enhancing the framework's scalability and practicality in large-scale or resource-constrained scenarios, making our defense even more robust.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62206238, 62406125), the Natural Science Foundation of Jiangsu Province (Grant No. BK20251911), and the China Postdoctoral Science Foundation (No. 2025T180438).

References

Bagdasaryan, E.; Veit, A.; Hua, Y.; Estrin, D.; and Shmatikov, V. 2020. How to backdoor federated learning. In *International conference on artificial intelligence and statistics*, 2938–2948. PMLR.

Blanchard, P.; El Mhamdi, E. M.; Guerraoui, R.; and Stainer, J. 2017. Machine learning with adversaries: Byzantine tolerant gradient descent. *Advances in neural information processing systems*, 30.

Borgwardt, K. M.; Ong, C. S.; Schönauer, S.; Vishwanathan, S.; Smola, A. J.; and Kriegel, H.-P. 2005. Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl_1): i47–i56.

Chen, X.; Liu, C.; Li, B.; Lu, K.; and Song, D. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. arXiv:1712.05526.

Dai, E.; Lin, M.; Zhang, X.; and Wang, S. 2023. Unnoticeable backdoor attacks on graph neural networks. In *Proceedings of the ACM Web Conference 2023*, 2263–2273.

Fung, C.; Yoon, C. J.; and Beschastnikh, I. 2020. The limitations of federated learning in sybil settings. In *23rd International symposium on research in attacks, intrusions and defenses (RAID 2020)*, 301–316.

Huang, M.; Liu, Y.; Ao, X.; Li, K.; Chi, J.; Feng, J.; Yang, H.; and He, Q. 2022. Auc-oriented graph neural network for fraud detection. In *Proceedings of the ACM web conference 2022*, 1311–1321.

Ju, W.; Fang, Z.; Gu, Y.; Liu, Z.; Long, Q.; Qiao, Z.; Qin, Y.; Shen, J.; Sun, F.; Xiao, Z.; et al. 2024. A comprehensive survey on deep graph representation learning. *Neural Networks*, 173: 106207.

Kipf, T. 2017. Semi-Supervised Classification with Graph Convolutional Networks. arXiv:1609.02907.

Li, K.; Shi, B.; Wei, J.; and Dong, B. 2025. NI-GDBA: Non-Intrusive Distributed Backdoor Attack Based on Adaptive Perturbation on Federated Graph Learning. In *Proceedings of the ACM on Web Conference 2025*, 852–862.

Li, Y.; Jiang, Y.; Li, Z.; and Xia, S.-T. 2022. Backdoor learning: A survey. *IEEE transactions on neural networks and learning systems*, 35(1): 5–22.

Li, Z.; Sharma, V.; and Mohanty, S. P. 2020. Preserving data privacy via federated learning: Challenges and solutions. *IEEE Consumer Electronics Magazine*, 9(3): 8–16.

Liu, Y.; Zhu, S.; Xia, J.; Ma, Y.; Ma, J.; Liu, X.; Yu, S.; Zhang, K.; and Zhong, W. 2024. End-to-end learnable clustering for intent learning in recommendation. *Advances in Neural Information Processing Systems*, 37: 5913–5949.

Lu, S.; Li, R.; Liu, W.; and Chen, X. 2022. Defense against backdoor attack in federated learning. *Computers & Security*, 121: 102819.

Mammen, P. M. 2021. Federated Learning: Opportunities and Challenges. arXiv:2101.05428.

Nguyen, T. D.; Rieger, P.; Chen, H.; Yalame, H.; Möllering, H.; Fereidooni, H.; Marchal, S.; Miettinen, M.; Mirhoseini, A.; Zeitouni, S.; et al. 2022. {FLAME}: Taming backdoors in federated learning. In *31st USENIX Security Symposium (USENIX Security 22)*, 1415–1432.

Ozdayi, M. S.; Kantarcioglu, M.; and Gel, Y. R. 2021. Defending against backdoors in federated learning with robust learning rate. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 9268–9276.

Rossi, R.; and Ahmed, N. 2015. The network data repository with interactive graph analytics and visualization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Shervashidze, N.; Schweitzer, P.; Van Leeuwen, E. J.; Mehlhorn, K.; and Borgwardt, K. M. 2011. Weisfeiler-lehman graph kernels. *Journal of Machine Learning Research*, 12(9).

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. arXiv:1710.10903.

Wan, G.; Shi, Z.; Huang, W.; Zhang, G.; Tao, D.; and Ye, M. 2025. Energy-based backdoor defense against federated graph learning. In *The Thirteenth International Conference on Learning Representations*.

Xie, C.; Chen, M.; Chen, P.-Y.; and Li, B. 2021. Crfl: Certifiably robust federated learning against backdoor attacks. In *International conference on machine learning*, 11372–11382. PMLR.

Xie, C.; Huang, K.; Chen, P.-Y.; and Li, B. 2019. Dba: Distributed backdoor attacks against federated learning. In *International conference on learning representations*.

Xu, J.; Wang, R.; Koffas, S.; Liang, K.; and Picek, S. 2022. More is better (mostly): On the backdoor attacks in federated graph neural networks. In *Proceedings of the 38th Annual Computer Security Applications Conference*, 684–698.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? arXiv:1810.00826.

Yang, Q.; Liu, Y.; Chen, T.; and Tong, Y. 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2): 1–19.

Yang, Y.; Li, Q.; Jia, J.; Hong, Y.; and Wang, B. 2024. Distributed backdoor attacks on federated graph learning and certified defenses. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, 2829–2843.

Zheng, H.; Xiong, H.; Chen, J.; Ma, H.; and Huang, G. 2023. Motif-backdoor: Rethinking the backdoor attack on graph neural networks via motifs. *IEEE Transactions on Computational Social Systems*, 11(2): 2479–2493.