Adaptive Privacy Defense Against Category Inference Attack in Clustered Federated Learning: Balancing Security and Model Performance

Cheng Gu, Lei Shi(✉), Binbin Liu, Hailong Tang, and Juan Xu

School of Computer Science and Information Engineering,
Hefei University of Technology, Hefei 230009, China
shilei@hfut.edu.cn

Abstract. Clustered Federated Learning (CFL) effectively addresses the challenge of data heterogeneity in Federated Learning (FL), where clients often hold Non-IID (Non-Independent and Non-Identically Distributed) data, typically limited to a few categories. However, the updates of the cluster models in CFL inadvertently expose additional information, rendering it vulnerable to Category Inference Attack (CIA), where the attacker exploits this exposure to infer sensitive category information from these updates. In our experiments on the image classification datasets, the attacker consistently achieves the F1-score exceeding 90% across various scenarios, highlighting CFL's vulnerability to CIA and the urgent need for robust privacy protections. To defend against this attack, we propose an adaptive local differential privacy (LDP) strategy for CFL, named AFC-CFL (Adaptive Fisher Information and Dynamic Clipping Threshold in CFL). AFC-CFL adopts adaptive Fisher information to adjust the privacy budget and dynamically modifies the clipping threshold during model training, mitigating the noise's effect on model performance while ensuring strong privacy protection. Experiments demonstrate that AFC-CFL significantly reduces the impact of noise on model accuracy, achieving a maximum accuracy improvement of 32.8% compared to common LDP method. Additionally, AFC-CFL reduces the attacker's F1-score by up to 24.3%, achieving a superior trade-off between model performance and privacy protection, making it highly suitable for deployment in privacy-sensitive CFL scenarios.

Keywords: Cluster federated learning · Category inference attack · Local differential privacy

1 Introduction

Clustered Federated Learning (CFL) [1] optimizes Federated Learning (FL) [2–4] by mitigating the impact of data heterogeneity through client clustering. However, despite only sharing model updates, CFL remains vulnerable to inference attacks [5,6] and backdoor attacks [7] that manipulate model behavior.

Z. Cai et al. (Eds.): WASA 2025, LNCS 15688, pp. 245–255, 2025.
https://doi.org/10.1007/978-981-96-8731-2_24

This paper focuses on category inference attacks (CIA) in CFL, where an attacker utilizes cluster similarity to deduce category information, posing serious privacy concerns. The clustering process further increases privacy risks by grouping clients with similar data, making it easier for attackers to infer category information. Our experiments show that the F1 score of CIA can exceed 90%. This indicates that an attacker can successfully infer the category information of individual clusters.

To defend against this attack, we propose an adaptive local differential privacy (LDP) [8–10] method for CFL, named Adaptive Fisher Information and Dynamic Clipping Threshold in CFL (AFC-CFL). AFC-CFL applies noise based on Fisher information to minimize accuracy loss and uses dynamic gradient clipping to handle parameter variations.

We validate AFC-CFL through experiments on MNIST, EMNIST, and CIFAR-10. Results show that our method significantly reduces the attacker's effectiveness while improving model accuracy compared to common LDP. This demonstrates AFC-CFL's ability to enhance privacy protection while maintaining strong model performance.

Our contributions are as follows:

- We analyze the CIA in CFL and reveal that attacker can infer category information with high F1-score, representing a significant privacy threat.
- We propose an adaptive LDP strategy that utilizes the Fisher information matrix to guide noise addition and employs a dynamic clipping threshold strategy to determine suitable clipping thresholds for each client.
- We validate our defense method through experiments in the image classification tasks, showing that our defense method achieves a superior balance between privacy protection and model accuracy.

2 Category Inference Attack in CFL

2.1 Threat Model

This section we first introduce the threat model. Figure 1 illustrates the CIA method within the CFL framework. The CFL algorithm is divided into the following three steps: **1)** The server distributes all cluster models to clients. **2)** Each client determines cluster i by selecting the cluster model with the lowest loss, updates its parameters, and uploads the updated parameters to the server. **3)** The server aggregates the uploaded parameters based on the clients' cluster.

In this CFL scenario, a central server collaborates with N clients to train P cluster models. Each client has a local dataset that may include multiple categories, and the number of categories in each cluster, comprising one or more clients, is denoted as X_j, where $j \in [P]$ and $X_j \geq 1$.

Within this threat model, we assume that one of the clients acts as an attacker. The attacker's objective is to accurately infer the category information contained within each cluster. The attacker is assumed to be honest-but-curious,

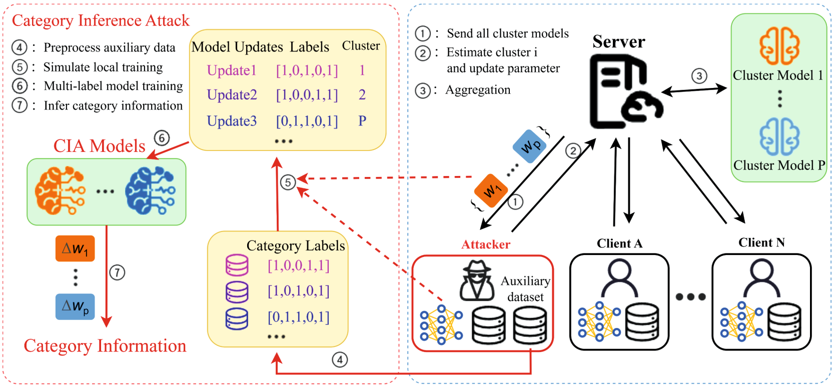


Fig. 1. Threat Model.

meaning it follows to the CFL protocol while actively attempting to extract additional information. Consequently, the attacker has access to all cluster models during each communication round.

Given that each cluster may include multiple categories ($X_j \geq 1$), the attacker trains a multi-label inference model for each cluster to achieve its objective. To train these multi-label inference models, the attacker collects task-relevant data from public sources to construct auxiliary datasets D_{aux}. The training process for the inference models involves three key steps, which are detailed in the subsequent subsection.

2.2 Attack Method

In this subsection, we describe the three steps an attacker follows to train the multi-label inference model: 1) Preprocess the auxiliary dataset: The attacker samples multiple subsets from the auxiliary dataset D_{aux}, constructing binary category labels l_{aux} to indicate the presence of specific categories. Repeating this process generates a dataset D_f for training the inference model. 2) Generate intermediate datasets: The attacker simulates the CFL process by evaluating the loss of each subset d_{aux} on cluster models w_j^t to determine its cluster. It then computes parameter updates $\Delta_{i,aux}$ and aggregates them into cluster-specific sets $G_{i,aux}$, forming intermediate training data. 3) Train the inference model: The attacker selects and aggregates parameter updates from $G_{i,aux}$, combining their labels via logical OR operations to generate the final training dataset $A_{j,aux}$. This dataset is used to train CIA models M_j for each cluster, enabling category inference from model updates.

Algorithm 1 outlines the framework for executing the three CIA processes. By using the difference between cluster model parameters from rounds t and $t+1$ ($\triangle w_j = w_j^{t+1} - w_j^t, j \in [P]$), the attacker uses the trained inference models to infer the category information for each cluster in round t, effectively revealing sensitive category-related information.

Algorithm 1. Category Inference Attack Models Training Algorithm

```
Input: Auxiliary dataset $D_{aux}$, Number of clusters $P$, Local learning rate $\eta$
Output: CIA Models $M_j, j \in [P]$
Initialize $D_f$, $G_{j,aux}, A_{j,aux}, j \in [P]$
while $|D_f|$ not big enough do
    $d_{aux}$ = Random sampling with replacement from $D_{aux}$
    $l_{aux}$ = Categories associated with the dataset $d_{aux}$
    $D_f = D_f \cup (d_{aux}, l_{aux})$
end while
Store all cluster models $w_j^t$, $j \in [P]$ in $t$-th round
for $(d_{aux}, l_{aux})$ in enumerate $D_f$ do
    cluster identity estimate $i = \arg\min_{j \in [P]} \nabla F(w_j^t, d_{aux})$
    $\Delta_{i,aux} = w_i^t$ - $\eta \nabla F(w_i^t, d_{aux})$
    $G_{i,aux} = G_{i,aux} \cup (\Delta_{i,aux}, l_{aux})$
end for
while $|A_{j,aux}|$ not big enough do
    $a_{j,aux}$ = the simulated aggregated updates from $G_{j,aux}$
    $L_{aux}$ = simultaneously merged $l_{aux}$ by logic OR operation.
    $A_{j,aux} = A_{j,aux} \cup (a_{j,aux}, L_{aux})$
end while
Train the inference model $M_j$ with $A_{j,aux}$
return $M_j, j \in [P]$
```

3 Adaptive Defense Method

In this section, we propose an innovative adaptive local differential privacy (LDP) defense strategy, called AFC-CFL (Adaptive Fisher Information and Dynamic Clipping Threshold in CFL), to reduce the threat of CIA in CFL.

Common local differential privacy methods typically apply the same amount of noise to each parameter. However, since each parameter has a different impact on model accuracy, adding the same amount of noise to all parameters can significantly degrade model performance. Additionally, using a fixed clipping threshold to limit the norm of the gradients may not be suitable for all clients, as different clients may have varying data distributions. Therefore, to address these challenges, we design an adaptive local differential privacy method called AFC-CFL based on LDP.

Firstly, to address the issue of significant accuracy degradation caused by adding the same noise to all parameters, we draw inspiration from the Fisher information matrix. By calculating the Fisher information for each parameter, we can determine the amount of information each parameter contains. Parameters with a larger amount of information have a greater impact on the model performance and clustering results. Consequently, we add less noise to parameters with higher Fisher information to minimize the impact of noise on model accuracy, while adding more noise to less informative parameters to enhance privacy protection.

For a client n with a private dataset D_n, where the model parameters trained in a particular round are denoted by $w_n = \{w_{n,1}, w_{n,2}, \ldots, w_{n,K}\}$ (with K being the total number of parameters), the Fisher information $I_{n,k}$ for each parameter $w_{n,k}$ is calculated as:

$$I_{n,k} = \mathbb{E}\left[\left(\frac{\partial \log L(w_n; D_n)}{\partial w_{n,k}}\right)^2\right]. \tag{1}$$

Here, $L(w_n; D_n)$ is the likelihood function of the data D_n given the parameter w_n. The Fisher information values are normalized using min-max normalization, as follows:

$$\hat{I}_{n,k} = \frac{I_{n,k} - \min\limits_k\{I_{n,k}\}}{\max\limits_k\{I_{n,k}\} - \min\limits_k\{I_{n,k}\}}. \tag{2}$$

To adaptively adjust the privacy budget for each parameter, we introduce the privacy budget ratio α_k. The privacy budget ratio is calculated as $\alpha_k = \beta - \hat{I}_{n,k}$, where β is a hyperparameter to control the overall privacy budget adjustment range. The privacy budget for each parameter is modified to $\epsilon_k = \epsilon/\alpha_k$. This dynamically adjusts the privacy budget for each parameter by calculating Fisher information value, which influences the noise magnitude applied to each parameter, thereby reducing the noise's impact on the model.

Second, due to the Non-IID nature of client data, gradient updates can vary significantly across clients, making a fixed gradient clipping threshold unsuitable for all. To address this, the client n dynamically determines its clipping threshold C_n^t by selecting the p-th percentile of its historical gradient L_2-norm values. Specifically, this is represented as $C_n^t = Percentile_p[G_n^0, G_n^1, \ldots, G_n^t]$, where G_n^t denotes the L_2-norm of the gradients at round t. This method allows each client to adaptively adjust its clipping threshold based on historical gradient statistics, ensuring the threshold aligns with its specific data distribution and gradient characteristics.

Overall, we combined the above two strategies to design the AFC-CFL algorithm, a comprehensive defense strategy against CIA in CFL environments. The entire process is outlined in Algorithm 2. In each training round, the server broadcasts all cluster models to the client for local training. The client then follows the steps below to execute the AFC-CFL algorithm and ensure the effectiveness of the adaptive defense strategy. First, the client determines its cluster i by selecting the cluster model with the lowest loss and updates its corresponding parameters. Next, the client computes the L_2-norm of the gradient and uses the dynamic clipping threshold strategy to update the clipping threshold, which is used to clip the gradient. In addition, the client computes the Fisher information for each parameter, calculates the privacy budget ratio, and modifies the privacy budget for each parameter.

The Gaussian Mechanism is applied to add random noise to the model parameters, following $\mathcal{N}(0, (\Delta f)^2 \sigma_k^2 \mathbb{I})$ distribution, where Δf is the sensitivity, which is calculated as $2\eta C/|b|$, with η is the learning rate, C is the current gradient

Algorithm 2. AFC-CFL

```
Input: Number of clusters $P$, Number of clients $N$, Local learning rate $\eta$, Global round
    $T$, Private dataset $D$, Initialization $w_j^0$, $j \in [P]$, Privacy budget $\epsilon$, Percentage of
    clip $p$, Adjustment of noise parameter $\beta$
Output: All cluster models $w_j^T$, $j \in [P]$
for each global round $t$ = 0, 1, 2,..., $T$-1 do
    Server: Broadcast $w_j^t$, $j \in [P]$ to all clients
    for each client $n \in N$ in parallel do do
        cluster identity estimate $i = \arg\min_{j \in [P]} F_n(w_j^t)$
        mini-batch $b_n$ sampled from $D_n$
        Compute the gradient $g_{n,x}^t = \nabla F(w_i^t, x)$, where $x \in \{1, ..., |b_n|\}$
        Calculate the local gradient paradigm $G_n^t = \frac{1}{|b_n|} \sum_{x \in b_n} \|g_{n,x}^t\|_2$
        Change clipping threshold $C_n^t = [G_n^0, G_n^1, , , G_n^t]_p$
        gradient clipping $g_n^t = \frac{1}{|b_n|} \sum_{x \in b_n} g_{n,x}^t / max(1, \frac{\|g_{n,x}^t\|_2}{C_n^t})$
        Calculate fisher information $\hat{I}_{n,k}$ by Eq. 1 and Eq. 2
        Privacy budget ratio $\alpha_k = \beta - \hat{I}_{n,k}$
        Change each parameter's privacy budget $\epsilon_k = \epsilon / \alpha_k$
        Update the model parameter $w_n^{t+1} = w_i^t - \eta g_n^t$
        Add noise for each parameter $\tilde{w}_{n,k}^{t+1} = w_{n,k}^{t+1} + \mathcal{N}(0, (\Delta f)^2 \sigma_k^2 \mathbb{I})$
        Send $\tilde{w}_n^{t+1}$ to Server
    end for
    Server: Aggregate parameters by clients's cluster identity estimates.
end for
return $w_j^T$, $j \in [P]$
```

clipping threshold, and $|b|$ is the size of the batch data b. For each parameter k, the noise parameter σ_k is given by Eq. 3:

$$\sigma_k = \frac{1}{\epsilon_k} \sqrt{2 \ln(\frac{1.25}{\delta})}. \tag{3}$$

Here, ϵ_k represents the privacy budget for each parameter, adjusted based on Fisher information, and δ refers to adjacent failure probability. At last, the noise of the appropriate scale size is added to the updated parameters, which are then sent to the server. With this adaptive defense, AFC-CFL effectively reduces the F1-score of CIA, thus enhancing the privacy protection in CFL while maintaining model performance.

4 Experiments

4.1 Experimental Settings

Datasets and Models. We evaluate our method on three widely used datasets: MNIST, EMNIST, and CIFAR-10. To simulate a realistic CFL, we adopt a Non-IID data partitioning scheme similar to [2]. The datasets are divided into sorted shards, with each client assigned two shards. This ensures that each client has data from only a limited number of categories. For the MNIST and EMNIST datasets, we utilize the Fully Connected Neural Network model, consisting of two fully connected layers activated with Leaky ReLU. In contrast, for the CIFAR-10 dataset, we employ the Convolutional Neural Network model featuring two convolutional layers followed by three fully connected layers, all activated with Leaky ReLU.

Experiment Setup. Each client updates its local model parameters using SGD (Stochastic Gradient Descent) and honestly uploads the updated parameters to the server. For MNIST and EMNIST, the learning rate is $\eta = 0.1$, and the model is trained for $T = 100$ rounds. And for CIFAR-10, the learning rate is $\eta = 0.05$, with a total of $T = 200$ training rounds. The batchsize for MNIST and EMNIST is set to 600, and for CIFAR-10, the batch size is set to 100. In the attack experiments, we evaluate the impact of varying the number of clusters and the number of clients on the effectiveness of attacks. The auxiliary dataset comprises approximately 3% to 5% of the training dataset. To train the inference model, the attacker samples $\gamma = 10,000$ small independent datasets. For the defense experiments, we fix the setting where the total number of clients participating in the global training process is $N = 10$, divided into $P = 2$ clusters. The value of the hyperparameter β is set to 1.3. Additionally, the adaptive clipping ratio p is fixed at 70%, and δ is set to 10^{-5}. All experiments are performed using PyTorch on an NVidia RTX A10 (24 GB) server.

Metrics. We evaluate the attack's accuracy and the defense's effectiveness using precision, recall, and F1-score. To calculate these metrics, we randomly select certain training rounds and calculate the mean precision ($\bar{p}$) and mean recall ($\bar{r}$) across all clusters. The F1-score is calculated as follows:

$$F = \frac{2\bar{p}\bar{r}}{\bar{p} + \bar{r}}. \tag{4}$$

4.2 Attack Result Analysis

The experimental results summarized in Table 1 clearly underscore the robustness of CIA in CFL scenarios. The attack consistently achieves high F1-score across the MNIST, EMNIST, and CIFAR-10 datasets, exceeding 90% in all tested configurations. This highlights the effectiveness of CIA in inferring category information under varying numbers of clients and clusters. The results further demonstrate the versatility of the attack method, proving its capability to operate effectively across different data distribution scenarios.

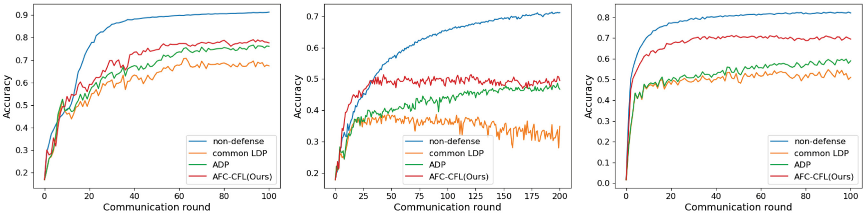
Notably, the attack's performance is influenced by the distribution of categories among cluster models. In scenarios with minimal or no category overlap between cluster models, the F1-score are significantly higher. As the category overlap decreases, the loss differences between clusters for a given small dataset

Table 1. The Attack Metrics

Dataset	Clusters P	Clients N	Precision	Recall	F1-score
MNIST	2	10	0.985	0.894	0.937
		15	0.999	0.886	0.939
		20	0.992	0.873	0.929
	3	15	0.976	0.867	0.918
		20	0.989	0.846	0.912
CIFAR-10	2	10	0.942	0.982	0.962
		15	0.984	0.921	0.951
		20	0.920	0.964	0.941
	3	15	0.946	0.869	0.906
		20	0.868	0.963	0.913
EMNIST	2	10	0.930	0.979	0.954
		15	0.965	0.914	0.939
		20	0.956	0.957	0.956
	3	15	0.883	0.937	0.909
		20	0.874	0.939	0.905

become more pronounced, making it easier to discern the category information associated with each cluster. For instance, under low category overlap, the F1-score on CIFAR-10 reaches up to 96.2% compared to 90.6% when overlap is high. Overall, the experimental results emphasize the need for enhanced privacy-preserving mechanisms in CFL to mitigate the risks posed by such inference attacks.

4.3 Defense Result Analysis



(a) MNIST (b) CIFAR-10 (c) EMNIST

Fig. 2. Comparison of the model accuracy in CFL between non-defense, common LDP, ADP and our AFC-CFL.

We conducted defense experiments in the attack scenarios described in Sect. 4.2, ensuring the same number of attack rounds for each scenario in the sim-

Table 2. The Attack Metrics and Model Accuracy in Various Scenarios

Scenario	Precision	Recall	F1-score	Model Accuracy
MNIST with non-defense	0.985	0.894	0.937	0.913
MNIST with common LDP	0.940	0.669	0.782	0.708
MNIST with ADP [9]	0.923	0.757	0.832	0.767
MNIST with AFC-CFL(Ours)	0.914	0.669	**0.773**	**0.790**
CIFAR-10 with non-defense	0.942	0.982	0.962	0.713
CIFAR-10 with common LDP	0.814	0.834	0.824	0.387
CIFAR-10 with ADP [9]	0.826	0.895	0.859	0.486
CIFAR-10 with AFC-CFL(Ours)	0.795	0.834	**0.814**	**0.514**
EMNIST with non-defense	0.930	0.979	0.954	0.825
EMNIST with common LDP	0.671	0.816	0.736	0.548
EMNIST with ADP [9]	0.678	0.833	0.748	0.598
EMNIST with AFC-CFL(Ours)	0.631	0.821	**0.722**	**0.712**

ulations. As shown in Table 2, our proposed defense method, AFC-CFL, demonstrates exceptional effectiveness by significantly reducing the attacker's scores. Specifically, AFC-CFL achieves a 17.5%(MNIST), 15.4%(CIFAR-10), and 24.3%(EMNIST) reduction in the attacker's score. Figure 2 provides a detailed visualization of the model accuracy trends across all datasets.

To further validate the performance of AFC-CFL, we compared it with the common Local Differential Privacy (LDP) defense method, which lacks adaptive adjustment capabilities. By carefully tuning the privacy budget of the common LDP, we ensured a comparable defense effect. However, while the common LDP defense also reduces the attacker's score, it results in a significant drop in model accuracy. In contrast, our method strikes a better balance between privacy protection and model accuracy. Specifically, compared to the common LDP method, AFC-CFL improves model accuracy by 11.6%(MNIST), 32.8%(CIFAR-10), and 29.9%(EMNIST) while maintaining the same level of defense effectiveness.

We also compare our method with another LDP-based strategy, Adaptive Differential Privacy (ADP) [9], which introduces decaying Gaussian noise during training. ADP employs the same initial noise scale as common LDP but reduces it during training with a decay rate of $R = 0.995$. Experimental results indicate that while ADP achieves some improvement in model accuracy by reducing the added noise, its defense effectiveness is weaker compared to the common LDP. In contrast, AFC-CFL surpasses ADP in both model accuracy and defense effectiveness, providing stronger privacy protection.

Overall, AFC-CFL achieves an excellent trade-off between model accuracy and privacy preservation. Not only does it maintain high model performance across different datasets, but it also significantly reduces the attacker's ability to infer category information, making it a robust defense in CFL settings.

5 Conclusion

Our research demonstrates that CIA represents a significant threat to CFL. Due to the inherent structure of CFL, detecting category information at the cluster level can indirectly reveal category information about the individual customers within that cluster. To address this threat, we propose a novel adaptive local differential privacy defense strategy called AFC-CFL, which strikes a better trade-off between privacy protection and model accuracy. In the future, we plan to broaden the scope of our experiments to cover a wider range of scenarios and datasets, with a particular focus on implementing CIA attacks and defenses in dynamic CFL environments. These dynamic settings may provide additional contextual information that can be utilized in conjunction with inference results so that private information about the target customer can be inferred. Furthermore, we aim to develop a robust defense framework that enhances privacy protection within collaborative learning systems.

Acknowledgments. This article was supported by Anhui Provincial Natural Science Foundation (Grant No. 2308085MF212 and No. 2308085MF221) and Joint Fund Project of Natural Science Foundation of Anhui Province (Grant No. 2208085US05 and No. 2308085US05).

References

1. Ghosh, A., Chung, J., Yin, D., Ramchandran, K.: An efficient framework for clustered federated learning. IEEE Trans. Inf. Theory **68**(12), 8076–8091 (2022)
2. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.Y.: Communication-efficient learning of deep networks from decentralized data. In: Singh, A., Zhu, J. (eds.) Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, vol. 54, pp. 1273–1282 (2017)
3. Shen, R., et al.: BAFL-SVM: a blockchain-assisted federated learning-driven SVM framework for smart agriculture. High-Confid. Comput. 100243 (2024)
4. Pang, J., Huang, Y., Xie, Z., Han, Q., Cai, Z.: Realizing the heterogeneity: a self-organized federated learning framework for IoT. IEEE Internet Things J. **8**(5), 3088–3098 (2021)
5. Gao, J., et al.: Secure aggregation is insecure: category inference attack on federated learning. IEEE Trans. Dependable Secure Comput. **20**(1), 147–160 (2023)
6. Yu, D., Zhang, H., Huang, Y., Xie, Z.: Data distribution inference attack in federated learning via reinforcement learning support. High-Confid. Comput. **5**(1), 100235 (2025)
7. Xu, H., Cai, Z., Xiong, Z., Li, W.: Backdoor attack on 3D grey image segmentation. In: 2023 IEEE International Conference on Data Mining (ICDM), pp. 708–717 (2023)
8. He, Z., Wang, L., Cai, Z.: Clustered federated learning with adaptive local differential privacy on heterogeneous IoT data. IEEE Internet Things J. **11**(1), 137–146 (2024)

9. Zhang, X., Ding, J., Wu, M., Wong, S.T.C., Van Nguyen, H., Pan, M.: Adaptive privacy preserving deep learning algorithms for medical data. In: 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1168–1177 (2021)
10. Xiong, Z., Cai, Z., Takabi, D., Li, W.: Privacy threat and defense for federated learning with non-I.I.D. data in AIoT. IEEE Trans. Ind. Inform. **18**(2), 1310–1321 (2022)