Unveiling Client Privacy Leakage from Public Dataset Usage in Federated Distillation

Haonan Shi
haonan.shi3@case.edu
Case Western Reserve University
Cleveland, Ohio, USA

Tu Ouyang
tu.ouyang@case.edu
Case Western Reserve University
Cleveland, Ohio, USA

An Wang
an.wang@case.edu
Case Western Reserve University
Cleveland, Ohio, USA

Abstract

Federated Distillation (FD) has emerged as a popular federated training framework, enabling clients to collaboratively train models without sharing private data. Public Dataset-Assisted Federated Distillation (PDA-FD), which leverages public datasets for knowledge sharing, has become widely adopted. Although PDA-FD enhances privacy compared to traditional Federated Learning, we demonstrate that the use of public datasets still poses significant privacy risks to clients' private training data. This paper presents the first comprehensive privacy analysis of PDA-FD in presence of an honest-but-curious server. We show that the server can exploit clients' inference results on public datasets to extract two critical types of private information: label distributions and membership information of the private training dataset. To quantify these vulnerabilities, we introduce two novel attacks specifically designed for the PDA-FD setting: a label distribution inference attack and innovative membership inference methods based on Likelihood Ratio Attack (LiRA). Through extensive evaluation of three representative PDA-FD frameworks (FedMD, DS-FL, and Cronus), our attacks achieve state-of-the-art performance, with label distribution attacks reaching minimal KL-divergence and membership inference attacks maintaining high True Positive Rates under low False Positive Rate constraints. Our findings reveal significant privacy risks in current PDA-FD frameworks and emphasize the need for more robust privacy protection mechanisms in collaborative learning systems.

1 Introduction

In recent years, federated learning (FL) has emerged as a promising paradigm for collaborative machine learning while preserving data privacy [26]. Traditional FL frameworks, such as FedAvg [26] and FedSGD, require clients to upload model parameters or gradients to a central server for aggregation, which can introduce limitations in both privacy and utility. To address these issues, federated distillation (FD) [3, 15, 17, 23, 35] has gained attention as an alternative approach that offers enhanced privacy protection and reduced communication overhead [14, 41]. In FD, model-inference outputs or distilled knowledge are exchanged between the server and clients instead of model parameters. This learning scheme only requires black-box access to client models, supporting diverse model architectures across clients. Existing approaches, such as FedMD [23], DS-FL [15] and Cronus [3], have been proposed to further enhance the privacy protection and efficiency of collaborative learning.

In these solutions, public datasets are often used to facilitate knowledge distillation among clients. Knowledge sharing can be achieved across clients with diverse data distributions by having all clients perform inference on the same public data and sharing these inference results as knowledge. We call such a learning scheme public dataset-assisted federated distillation, or PDA-FD. Despite the benefits of FD, the privacy implications in such frameworks have not been thoroughly explored in the existing literature. While FD generally provides stronger privacy guarantees than traditional FL, using a public dataset for information exchange can result in potential privacy leakage. Figure 1 illustrates the workflow of PDA-FD. In each collaborative training round of PDA-FD, clients need to train their private models on their respective private datasets before performing inference on the public dataset and transmitting the inference results as knowledge to the server. This process enables better transfer of knowledge learned from private data to other clients' private models. However, this process also enhances the memorization of private datasets by private models, thereby causing the private models' inference results on the public dataset to leak private information from the private dataset. We discover that an *honest-but-curious* server, via manipulating the public dataset and exploiting the client' inference results on the public dataset, can obtain private information from a particular client's private training dataset without compromising the training process of PDA-FD. In particular, the label distribution of the training dataset, and if specific data belongs to a client's training dataset (membership information). These two information leaks are representative and frequently studied in the ML privacy literature [10, 28, 43].

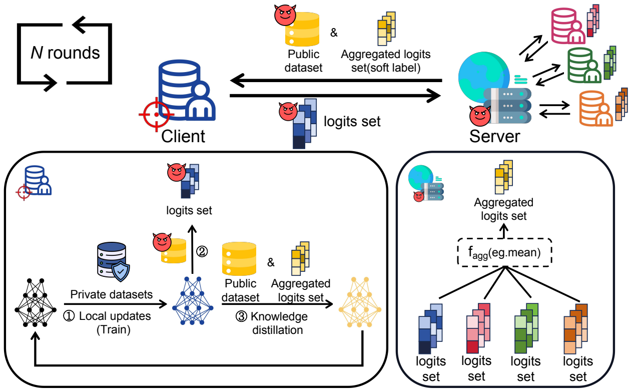


Figure 1: Workflow of Public Dataset-Assisted Federated Distillation (PDA-FD). During each collaborative training round, clients first train their private models on their respective private datasets, then perform inference on the public dataset and transmit the resulting logits back to the server as transferred knowledge. As an *honest-but-curious* server, the server can extract private information from target clients' private datasets by manipulating and leveraging public datasets.

In this work, we devise two types of privacy attacks against clients by the server: *Label Distribution Inference Attacks (LDIA)* and *Membership Inference Attacks (MIA)*. LDIA reveals privacy information about the overall data distribution of all client private datasets combined. In contrast, membership inference attacks enable more granular privacy leakage by identifying the presence of specific data samples in a client's private datasets. Both LDIA and MIA can be performed in a black-box manner, requiring only clients' inference results on the public dataset, which renders these attacks practically useful since FD frameworks are generally designed to not transfer a client model to the server. Additionally, Federated Distillation assumes non-IID data distributions across clients in the label space. LDIA becomes particularly important in this context, as inferring the label distribution can reveal significant information about a client's unique data characteristics. Furthermore, LDIA and MIA can serve as stepping stones for more sophisticated attacks. For instance, the obtained knowledge can be leveraged to generate synthetic data that mimics private datasets.

In the recent literature, LDIA and MIA have been extensively studied in the traditional FL and centralized machine learning settings [19, 27, 28, 30, 34, 44]. They have largely focused on white-box or gradient-based attacks. For example, Gu *et al.* demonstrated that a malicious server in FL can infer label distributions by exploiting the gradients or parameters uploaded by clients [10]. Similarly, Wainakh *et al.* showed that user-level label leakage is possible through gradient analysis [37]. For MIAs, Nasr *et al.* developed sophisticated MIA techniques that exploit the white-box access to model parameters in FL [28]. The rich information leveraged in these attacks are not directly accessible in FD settings. In the FD setting, a few studies have primarily focused on MIAs, such as FD-Leaks [43], MIA-FedDL [24] and GradDiff [38]. In Fed-leaks, a malicious client leverages its local model as a shadow model to train an MIA classifier, then applies the classifier on target sample inference score to determine target sample membership in other clients' private datasets. The attacker needs to be the client in the *MIA-FedDL* approach; *GradDiff* can be used by either a client or the server to attack another client. Both of these approaches require shadow datasets to train shadow models for MIA. However, effective MIA requires the attacker's shadow model to be trained on the dataset sharing similar data distribution with the target model's training dataset. In the FD setting, it is challenging for both clients and the server to obtain the data distribution of other clients' private data. Therefore, these MIA methods cannot work effectively in heterogeneous non-IID environments.

To address the limitations of the existing works, we aim to comprehensively examine privacy leakage by public datasets in FD across multiple frameworks (FedMD, DS-FL, and Cronus) and various data distribution scenarios. We also introduce new attack methods that are specifically tailored to the PDA-FD setting. Specifically, we propose a novel LDIA method based on public datasets and extend the state-of-the-art MIA, Likelihood Ratio Attack (LiRA) [2], to overcome the challenges posed by the limited information available in PDA-FD. To that end, we design and implement *Co-op LiRA* and *Distillation-based LiRA* for MIAs. Furthermore, while previous works primarily focused on client-side attacks, our work provides a more holistic view by examining both LDIA and MIA from the server's perspective. A high-level comparison with the existing work is shown in Table 1.

Table 1: Summary of Privacy Attacks in FL & FD

Method	Attacker	Framework	Shadow dataset	Attack Goal
[10]	Server	FL	Required	LDIA
[28]	Server&Client	FL	Required	MIA
[24]	Client	FD	Required	MIA
[43]	Client	FD	Not Required	MIA
[38]	Server&Client	FD	Required	MIA
Ours	Server	FD	Not Required	LDIA&MIA

In our study, our key findings include: (1)LDIA can be successfully achieved across multiple PDA-FD frameworks that significantly outperform random guessing baselines. (2)The proposed *co-op LiRA* and *distillation-based LiRA* shows high effectiveness in terms of the True Positive Rate (TPR) in a low False Positive Rate (FPR) region. (3) We also show that the effectiveness of LDIA and MIA varies with data distributions. Adversaries can generally achieve higher attack success rates when data follows more uniform distributions compared to non-IID settings. Our findings collectively reveal significant privacy risks in current PDA-FD frameworks and highlight the need for more advanced privacy-preserving mechanisms.

2 Background

2.1 Federated Distillation

Federated Distillation (FD) [3, 15, 17, 23, 35] is a specialized FL framework distinct from traditional FL. FD exchanges model outputs or distilled knowledge between the server and clients instead of model parameters, which significantly reduces communication overhead. Additionally, FD only requires black-box access to client models and supports diverse model architectures across clients. As a result, FD not only better preserves privacy but also offers greater utility compared to traditional FL frameworks.

In our study, we focus on one category of FD frameworks, **Public Dataset-Assisted Federated Distillation** [3, 15, 23](PDA-FD). The workflow of PDA-FD is shown in Figure 1. In PDA-FD, the server leverages a public dataset to facilitate knowledge transfer among clients. The public dataset is shared with all the clients. The PDA-FD framework typically involves three phases in each collaborative training round: local updates phase, communication phase, and knowledge distillation phase.

During the local updates phase, client n trains its local model θ_n on its private dataset D_n using stochastic gradient descent [22]. The loss function $\mathcal{L}(x, y, \theta_n)$ is defined to calculate the error between the prediction posterior $f_{\theta_n}(x)_y$ of the training data and its ground truth label y. Cross-entropy is often used as the loss function:

$$\mathcal{L}(x, y, \theta_n) = -log(f_{\theta_n}(x)_y) \tag{1}$$

During the communication phase, a set of data samples S_t are selected by the server from the public dataset. For each selected sample x_k, the client's local model θ_n performs inference to obtain the corresponding logits $z_{\theta_n}(x_k)$ and sends the logits to the server. After collecting the logits from all clients, the server aggregates them using an aggregation algorithm f_{agg}(eg. average aggregation [23]) to get an aggregated logits Z_k. The server then distributes Z_k back to each client. At the end of the communication phase, each client will receive the logits set $\{Z_k \mid k \in S_t\}$ from server.

During the knowledge distillation phase, client n performs knowledge distillation [13] using the aggregated logits Z_k returned by the server as the soft labels. In this case, the loss function also needs to include the mean absolute errors (MAE):

$$\mathcal{L}(x, y, \theta_n) = \frac{1}{N}\sum_{n=1}^{N}\left|Z_k - z_{\theta_n}(x_k)\right| \tag{2}$$

or Kullback-Leibler (KL) divergence values [21]:

$$\mathcal{L}(x, y, \theta_n) = \sum_{n=1}^{N} Z_k \cdot \log\left(\frac{Z_k}{z_{\theta_n}(x_k)}\right) \tag{3}$$

The overall procedure of PDA-FD learning is summarized in Algorithm 1. Different PDA-FD frameworks [3, 15, 23] require clients to upload either logit vectors or prediction probability vectors during communication.

Algorithm 1 Public Dataset-Assisted Federated Distillation

```
Require: Private datasets $\{D_n\}_{n=1}^N$, public dataset $D_{pub}$, local models $\{\theta_n\}_{n=1}^N$, number of collaborative training round $T$, public data index set $\{S_t\}_{t=1}^T$.
for collaborative training round $t = 0$ to $T$ do
    ▷ Local Updates Phase
    Each client trains local model $\theta_n$ on private dataset $D_n$
    ▷ Communication Phase
    Each client computes logits $\{z_{\theta_n}(x_k) \mid k \in S_t\}$ for public data $\{x_k | k \in S_t\}$
    Each client sends logits set $\{z_{\theta_n}(x_k) \mid k \in S_t\}$ to server
    for $k \in S_t$ do
        $Z_k \leftarrow f_{agg}(\{z_{\theta_n}(x_k)\}_{n=1}^N)$
    end for
    Server sends aggregated logits $\{Z_k \mid k \in S_t\}$ to clients
    ▷ Knowledge Distillation Phase
    Each client trains $\theta_n$ on $\{x_k \mid k \in S_t\}$ using $\{Z_k\}$ as soft labels
end for
```

In this paper, we primarily focus on three PDA-FD frameworks: FedMD [23], DS-FL [15] and Cronus [3]. Each has a specific customization of the procedure demonstrated in Algorithm 1. In FedMD, each client n needs to train their local model θ_n on the public dataset D_{pub} until convergence before the collaborative training. In DS-FL, the server employs the entropy reduction aggregation (ERA)[15] algorithm as f_{agg} during the communication phase. ERA accelerates convergence and enhances the robustness of DS-FL in non-IID data distribution scenarios. In Cronus, the server utilizes the mean estimation algorithm proposed by Diakonikolas *et al.* [7] for logits aggregation algorithm f_{agg} to enhance robustness.

2.2 Label Distribution Inference Attack

In both FL and FD, Label Distribution Inference Attacks (LDIA) can pose significant threats to individual clients' privacy. In an LDIA, the adversary aims to infer the proportion of training data across different labels in the target client's training dataset. Previous work [10, 37] has demonstrated that the server of FL can infer the distribution of clients' training data by exploiting the gradients or parameters uploaded by clients in FL, leading to privacy leakage.

Definitions. Given a target client's local model θ_n that is trained on a private dataset D_n, which consists of data samples from M classes: $D_n = \bigcup_{m=1}^M D_n^m$, where D_n^m refers to the subset containing data of label m. The ground truth label distribution of D_n is calculated as follows:

$$\mathbf{p} = (p_1, p_2, \ldots, p_M), \quad p_m = \frac{|D_n^m|}{|D_n|} \tag{4}$$

where p_m represents the proportion of data with label m in D_n, and $\sum_{m=1}^M p_m = 1$ and $p_m \in [0, 1]$. The objective of LDIA can be expressed as a function $\mathcal{A}$ that maps from the observable information about the target model outputs to an estimated label distribution:

$$\mathcal{A} : \theta_n \mapsto \hat{\mathbf{p}} = (\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_M) \tag{5}$$

where $\hat{p}_m$ represents the inferred proportion of data with label m in D_n.

The success of LDIA can compromise the privacy of FL and FD systems, as it allows attackers to gain insights into the composition of private datasets without direct access.

2.3 Membership Inference Attack

In membership inference attacks (MIA), the attacker's objective is to determine whether a target data sample is in the target model's training dataset. As machine learning models become more widely deployed, their training datasets may contain sensitive information [9, 11], and MIA poses a significant threat to the privacy of such data. FD, which was designed to protect data privacy, has also become the target of various MIA attacks [38, 43] against clients' local models within the framework.

Definitions. Given a target model θ, and a target data sample x, the objective of MIA can be define as a function $\mathcal{A}$:

$$\mathcal{A} : x, \theta \mapsto \{0, 1\} \tag{6}$$

where the output is 1 if x is inferred to be in the training dataset of model θ, and 0 otherwise. In MIA, we refer to the data sample belonging to the target model's training set as members, while other data samples are referred to as non-members. MIA exploits the behavioral differences of machine learning models on data they were trained on versus data they weren't, leveraging the fact that models often exhibit higher confidence or lower loss on their training data due to overfitting.

3 Methodology

3.1 Threat Model

Adversary Knowledge: We investigate PDA-FD in the context of Horizontal FL (HFL), where clients' private data's label distributions can be non-IID. We consider a threat model where the server acts as an *honest-but-curious* [29] adversary. The server attacker is not allowed to modify the learning process but gets to select public dataset members used for knowledge transfer; this privilege for the server is common in PDA-FD frameworks [3, 15, 23]. The server can select members of the public dataset in each collaborative training round. In each training round, the server's access to clients' models is black, and it only interacts with client models by running inferences on public data. The server can only obtain logit vectors or prediction vectors of every public data sample each client model provides without visibility into the private models' weights or architectures.

Adversary's objective:. The server aims to infer sensitive information about a target client's private dataset, specifically label distribution information and membership information, through two main attack vectors: LDIA and MIA.

3.2 Label Distribution Inference Attack

Unlike traditional FL where the server has white-box access to clients' local models, the server only has black-box access by using a public dataset in PDA-FD. The core idea behind LDIA is that the logits or prediction values produced by a client's model still carry information about the distribution of labels in its training data. This is due to the tendency of neural networks to overfit their training distribution, even when regularization techniques are applied [44].

To validate this idea, we conduct a motivating experiment using the CIFAR-10 dataset [20]. We sample a subset of data with a non-IID label distribution from CIFAR-10 as the training dataset D_{train}. Subsequently, we train a deep neural network model θ on this training set. We use the trained model θ to infer on data x from the CIFAR-10 test dataset, which consists of ten different labels, to obtain the logits vector $z_\theta(x)$ from these inference results. We also apply the softmax function to get the posterior probabilities vector v_x of each image prediction:

$$v_{x,i} = \frac{e^{z_\theta(x)_i}}{\sum_{j=1}^{10} e^{z_\theta(x)_j}} \tag{7}$$

, where $v_{x,i}$ is the probability for the $i-$th label. For each label, we have 500 data samples for model inference and calculate the mean posterior probability vector across all the samples $V_{mean} = \frac{1}{|S|}\sum_{k\in S} v_{x_k}$, where S represents the subset for each label.

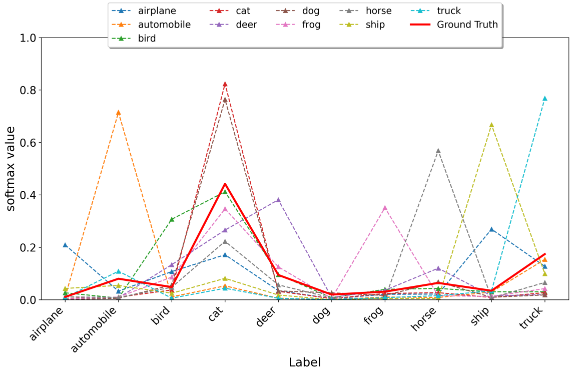


Figure 2: The mean vector of posterior probability vectors V_{mean} predicted by the model θ on 500 data samples with the same label in CIFAR10.

Figure 2 illustrates the mean posterior probability vector V_{mean} for predictions made by model θ on the subset of data for each label, alongside the label distribution of the training dataset D_{train} used for model θ. As seen in the figure, the labels "cat" and "truck" have a larger proportion among all labels in the training dataset D_{train}. However, they constantly appear with higher probabilities in the model's predictions, even for samples of other classes. For example, when the model is presented with "horse" images, the average prediction probabilities are 0.57, 0.22, and 0.07 for classes "horse", "cat" and "truck", respectively, which are the top-3 predictions. In this case, the model assigns higher probabilities to the over-represented classes, i.e., "cat" and "truck", over the other incorrect classes. This indicates that the model is able to retain the distribution of its training dataset to some extent. Such an observation suggests that in FD, where clients share logits or probability values of their models on public data samples, a malicious server could potentially infer the label distribution of clients' private training data. Even though the public data may have a different distribution, the clients' models will still exhibit biases reflective of their training data distributions.

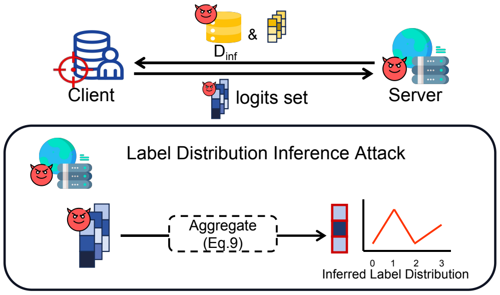

D_inf
&
Client
logits set
Server
Label Distribution Inference Attack
Aggregate
(Eq.9)
0
1
2
3
Inferred Label Distribution

Figure 3: Workflow of Label Distribution Inference Attack.

With all these positive verification experiment results, we design a LDIA method in PDA-FD that consists of the following steps: (1) As shown in Figure 3, the server selects a subset of the public dataset D_{pub} for each round of FD, which is denoted as D_{inf}. This selection needs to ensure that the samples are evenly distributed across different classes, which not only helps minimize bias in LDIA but also enhances FD's performance. (2) During the communication phase, each client performs inference on the selected public data samples $\{x_k \mid k \in S_t\}$. The generated logits are sent to the server. (3) The server receives logits from all the participating clients and selectively aggregate the logits sent by the target client. The server uses the resulting vector to infer the label distribution of the target client's private dataset.

Formally, the LDIA process can be expressed as:

$$\hat{p} = \frac{1}{\mid D_{inf} \mid} \sum_{x\in D_{inf}} softmax(z_\theta(x)) \tag{8}$$

, where $\hat{p}$ denotes the inferred label distribution, θ is the target model, and $z_\theta(x)$ represent the logits vector obtained by model θ when inferring on data x. To enhance accuracy and robustness, we average the inferred label distributions over N rounds and report the averaged distribution as the final LDIA result. This mitigates the impact of potential anomalies or fluctuations in individual rounds, thus improving overall stability.

3.3 Membership Inference Attack

MIA in machine learning aims to determine whether a specific data sample was used to train a model. Recent works have demonstrated the feasibility of MIA in traditional FL and centralized machine learning setting, where adversaries have either white-box access to the model's parameters and gradients or access to shadow datasets matching the target model's training dataset's data distribution for performing MIA [2, 34, 39]. However, in the context of PDA-FD, the server is limited to black-box interactions with the target client's model, and lacks knowledge of clients' private data distributions, making it challenging to obtain appropriate shadow datasets. To address this challenge, we propose adapting and enhancing existing

MIA approaches by combining them with the unique characteristics of the FD process.

Traditional MIA techniques often exploit the observation that machine learning models behave differently on data they are trained on versus data they aren't. The difference typically manifests as higher confidence or lower loss on training samples. In our proposed attack, we extend the state-of-the-art MIA technique, Likelihood Ratio Attack (LiRA), proposed by Carlini *et al.* [2].

Offline LiRA uses multiple reference ("out") models to establish a baseline prediction distribution for membership inference through hypothesis testing. The key assumption is that member samples have statistically different prediction scores between target and reference models, while non-members exhibit similar predictions across all models. For a target sample (x, y), LiRA first queries all reference models to obtain posterior probabilities, which are then standardized using the following scaling function:

$$\phi(f_\theta(x)_y) = \log\left(\frac{f_\theta(x)_y}{1 - f_\theta(x)_y}\right) \tag{9}$$

where $f_\theta(x)_y$ denotes the posterior probability from a reference model θ. The scaled scores from reference models are fitted to a Gaussian distribution $\mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}})$. The membership probability λ is then computed by comparing the target model's scaled score $\phi(f_{\theta_t}(x)_y)$ with this distribution:

$$\lambda = 1 - \Pr[Z > \phi(f_{\theta_t}(x)_y)], \text{where } Z \sim \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}}). \tag{10}$$

A higher scaled score $\phi(f_{\theta_t}(x)_y)$ relative to μ_{out} indicates a higher probability of (x, y) being a member of the training dataset.

While LiRA is a powerful and effective technique for MIAs, directly applying it in the context of PDA-FD presents challenges. In traditional LiRA, the attacker needs the ability to train multiple reference models that mimic the target model's behavior but behave differently on the target samples, specifically, prediction discrepancies between target and reference models should be larger for members than for non-members. However, training such models requires access to a shadow dataset with a distribution similar to the target model's training data [2, 39]. As shown in Figure 5a, when the data distribution of the reference model's training dataset differs from that of the target model's training dataset, the attacker cannot distinguish members based on prediction discrepancies between reference and target models for members versus non-members. The prediction discrepancy is quantified as the difference between the target model's prediction probability and the mean of reference models' prediction probabilities for the target sample. In the PDA-FD context, the server faces significant challenges in obtaining such a shadow dataset: (1) The FD framework allows clients to have non-IID private datasets in the label space. Without knowledge of the specific label distributions in each client's private dataset, the server cannot accurately sample shadow datasets that match the characteristics of the target client's data. (2) In FD, the public dataset can be unlabeled, particularly in semi-unsupervised learning scenarios. In that case, the server simply cannot use these datasets without labels for training shadow models, as label information is essential for mimicking the target model's behaviors. To address these unique challenges, as shown in Figure 4, we design and implement two variants of offline LiRA: Co-op LiRA and Distillation-based LiRA.

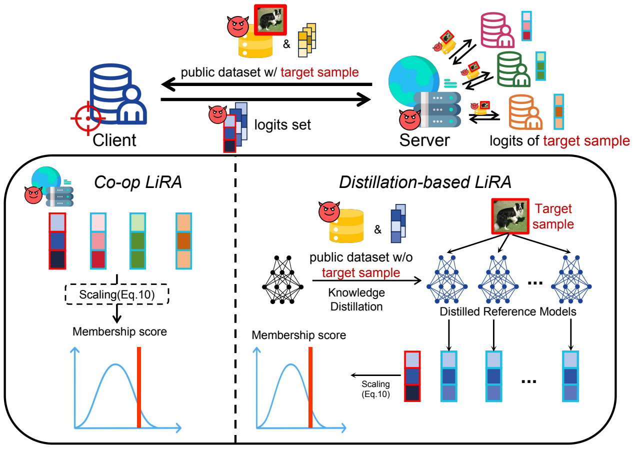
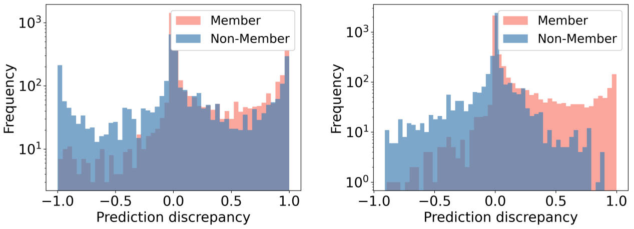


Figure 4: Workflow of Membership Inference Attack(Co-op LiRA and Distillation-based LiRA).

Co-op LiRA. Co-op LiRA is designed for scenarios where clients' private datasets exhibit similar data distributions. In this case, the server can utilize the non-target clients' local models as reference models to conduct MIA on the target client's model. We notice that when target samples appear in other clients' private datasets besides the target client, these clients' private models cannot serve as reference models. However, in real-world FD deployments, clients are typically assumed to have mostly disjoint training datasets to benefit collaborative learning, thus making shared target samples rare. For Co-op LiRA, the server can conduct the proposed LDIA described in Section 3.2 to determine label distributions across all clients first. In the HFL setting, similar label distributions across clients indicate similar overall data distributions, resulting in comparable performance of local models. Then, the server can use other clients' local models as reference models for LiRA directly based on the target client model. Algorithm 2 outlines the co-op LiRA process in PDA-FD: (1) During the communication phase, the server inserts the target sample into the selected public data subset to obtain the posterior probabilities from all clients' models, including the target clients. (2) The server conducts LDIA on all clients. (3) The server calculates the KL-divergence between the label distributions of the target client and each remaining client, selecting clients as reference models when their KL-divergence falls below threshold β (0.1). (4) The server performs LiRA hypothesis test by comparing reference models' and target client's posterior probabilities to determine target sample's membership probability.

Co-op LiRA eliminates the need for the attacker to train multiple reference models. However, the drawback of the method is that it becomes ineffective when very few clients have training data of the same or similar distribution to the target client.

Distillation-based LiRA. To address the limitation of co-op LiRA, we proposed another extension of LiRA called distillation-based LiRA. It is challenging for the server to obtain a shadow dataset which has same data distribution with the target's model's training dataset to train a reference model; we instead choose to use knowledge distillation to generate a student model, which is then used as a reference model for the MIA. On the one hand, the generated



(a) Reference model trained with D_{ref}. **(b) Reference model distilled by using D_{ref}.**

Figure 5: Prediction discrepancies between target and reference models across members and non-members on CIFAR 10. The target model is trained on dataset D_{target} while the adversary can only obtain D_{ref}, where D_{target} and D_{ref} have different distributions.

Algorithm 2 Co-op LiRA and Distillation-based LiRA. The shadow model preparation phase is implemented in lines 6-12 for Co-op LiRA and lines 14-18 for Distillation-based LiRA.

```
Require: public dataset $D_{pub}$, target sample $(x, y)$, target client
    private model $\theta_{target}$, clients private models $\{\theta_n\}_{n=1}^N$, scaling
    function $\phi$, KL-divergence threshold $\beta$, number of reference
    models $K$.
 1: $\mathcal{M}_{ref} \leftarrow \{\}, Conf \leftarrow \{\}$
 2: ▷ Communication phase in FD
 3: $D_{pub} \leftarrow D_{pub} \cup \{(x, y)\}$
 4: Server send $D_{pub}$ to all clients and receive $\{f_{\theta_n}(D_{pub})\}_{n=1}^N$
 5: ▷ Co-op LiRA
 6: $\hat{p}_{target} \leftarrow \mathrm{LDIA}(\theta_{target})$
 7: for $\theta_n \in \{\theta_n\}_{n=1}^N \setminus \{\theta_{target}\}$ do
 8:     $\hat{p}_n \leftarrow \mathrm{LDIA}(\theta_n)$
 9:     if $\mathrm{KL}(\hat{p}_{target}, \hat{p}_n) < \beta$ then
10:         $\mathcal{M}_{ref} \leftarrow \mathcal{M}_{ref} \cup \{\theta_n\}$
11:     end if
12: end for
13: ▷ Distillation-based LiRA
14: $D_{distill} \leftarrow$ Random sample$(D_{pub} \setminus \{(x, y)\})$
15: for $k = 1$ to $K$ do
16:     $\theta_k \leftarrow \mathrm{Distill}(\theta_{target}, D_{distill})$
17:     $\mathcal{M}_{ref} \leftarrow \mathcal{M}_{ref} \cup \{\theta_k\}$
18: end for
19: for $\theta \in \mathcal{M}_{ref}$ do
20:     $Conf \leftarrow Conf \cup \{\phi(f_\theta(x)_y)\}$
21: end for
22: $\mu_{out} \leftarrow \mathrm{mean}(Conf), \sigma_{out}^2 \leftarrow \mathrm{var}(Conf)$
23: return $\lambda \leftarrow 1 - \Pr[Z > \phi(f_{\theta_{target}}(x)_y)]$, where $Z \sim$
    $\mathcal{N}(\mu_{out}, \sigma_{out}^2)$
```

student model should behave similarly to the target model, exhibiting close prediction scores on non-member samples. On the other hand, it produces different prediction scores on member data that differ from those from the teacher model, since the student model is trained through knowledge distillation without direct exposure to the training dataset, whereas the teacher model, trained directly on member samples, exhibits a degree of overfitting to these samples. Previous research on MIA [16] has demonstrated that the student model generated through knowledge distillation can potentially encode membership information of the teacher model. However, we find significant prediction discrepancies between student and teacher models when performing inference on the teacher model's member samples. As shown in Figure 5b, the prediction discrepancies between the teacher and the student model are significant for some members while remaining relatively small for non-members. This distinct pattern enables these distilled student models to serve as effective reference models for offline LiRA. These reference models allow attackers to identify members with high prediction discrepancies while maintaining a low False Positive Rate, thus achieving high True Positive Rate (TPR) at low False Positive Rate (FPR), which serves as a critical performance metric for successful membership inference attacks[2, 39]. We validate this characteristic of knowledge distillation-based reference models in Section 4.3.

Algorithm 2 outlines the distillation-based LiRA process in PDA-FD: (1) During the communication phase, the server inserts the target samples into the public dataset. The server obtains the posterior probabilities predicted by the target client on all the data samples. These probability values could be used as soft labels to distill reference models. (2) To create multiple reference models, the server randomly samples the distillation dataset of distilled model from the public dataset except the target samples. Using the target client as the teacher model, the server distills multiple reference models. (3) The server then infers membership of the target samples by performing LiRA using the created reference models. This approach is more robust to heterogeneous data distributions among clients but comes at the cost of additional computational overhead for the distillation process.

4 Evaluations

In this section, we conduct a series of experiments to evaluate the privacy leakage in PDA-FD by conducting the proposed LDIA and MIA. Our experiments span multiple datasets, various PDA-FD frameworks [3, 15, 23], and different usage scenarios. Our experiments demonstrate four key aspects: (1) the overall effectiveness of our proposed LDIA and MIA; (2) the impact of varying non-IID data distributions; (3) the impact of different PDA-FD frameworks; and (4) the effect of the number of collaborative training rounds.

4.1 Experiment Setup

4.1.1 Datasets In our experiments, we utilize the following image datasets that are commonly used to test the performance of different FD frameworks: CIFAR-10[20], CINIC-10[6], Fashion-MNIST[42]. Additionally, for the completeness of the experiments, we also use a tabular dataset:Purchase[8]. In our experiments, we partition each dataset into a 4:1 ratio for clients' training sets D_{train} and the public dataset D_{pub}. To align with the previous FD frameworks [23, 43], we also configure a scenario where there is a data distribution discrepancy between the public dataset and the clients' private datasets. In this scenario, we use CIFAR-10 for client training and CIFAR-100 as the public dataset to simulate distribution shifts. Table 2 details the specific partitioning of datasets and the number of classes used in our experiments. In our MIA experiments, to ensure adequate private data for each client, we select an equal number of samples from the test dataset to serve as non-members.

Table 2: Datasets division.

Datasets	number of classes	D_{train}	D_{pub}	D_{test}
CIFAR-10	10	40000	10000	10000
CIFAR-10/CIFAR-100	10	40000	10000	10000
CINIC-10	10	72000	18000	90000
Fashion-MNIST	10	48000	12000	10000
Purchase	10	21589	5397	11565

In our experiments, we create 10 clients that participate in the collaborative training. To partition the training dataset D_{train} into private datasets D_n for 10 clients, we use Dirichlet distribution $Dir(\alpha)$ with α values of 0.1, 1 and 10 to generate non-IID data distribution across all the clients. The smaller the value of α, the more imbalanced the label distribution of D_n is.

4.1.2 Models. For different datasets, the clients in PDA-FD use different model architectures. When the private dataset is CIFAR-10, the clients train the ResNet-18 models [12]. For CINIC-10, the clients train the MobileNetV2 models [32]. For Fashion-MNIST datasets, the clients' local models employ a CNN architecture with four convolutional layers. When training with the Purchase dataset, the clients train MLP models consisting of three fully connected layers. As the heterogeneity in client model architectures does not affect our attack methodology[2], we adopted identical model structures across all clients.

4.1.3 LDIA Metrics. To evaluate the effectiveness of the proposed LDIA, we adopt the same metrics employed in previous LDIA research [10, 31]. In the equations for calculating these metrics, $\hat{p}$ denotes the inferred label distribution, p denotes the ground truth label distribution, m represents a specific label, and M denotes the number of labels:

- **Kullback-Leibler divergence.** Kullback-Leibler divergence(KL divergence) between the ground truth label distribution and the inferred label distribution can be calculated using the following equation:

$$Dis_{KL-div}(\hat{p},p) = \sum_{m=1}^{M} \hat{p}_m log(\frac{\hat{p}_m}{p_m}) \tag{11}$$

  This metric represents the similarity between the inferred label distribution and the ground truth label distribution. A smaller KL divergence indicates that the two distributions are more closely aligned.
- **Chebyshev distance.** The Chebyshev distance represents the maximum error between the inferred label distribution and the ground truth label distribution for each target client in an LDIA:

$$Dis_{Cheb}(\hat{p},p) = max_m \mid \hat{p}_m - p_m \mid \tag{12}$$

  A smaller Chebyshev distance indicates a higher reliability of the LDIA results.
- **Mean $l1$-distance.** The mean $l1$-distance represents the potential error between the inferred label distribution and the ground truth label distribution across all classes:

$$Dis_{mean-l1}(\hat{p},p) = \frac{1}{M} \sum_{m=1}^{M} \mid \hat{p}_m - p_m \mid \tag{13}$$

  A smaller mean $l1$-distance indicates a higher average accuracy of the LDIA results.

4.1.4 MIA Metrics. Same as previous efforts on MIA [2, 34, 39] , we employ the following metrics to evaluate the effectiveness of the proposed MIAs:

- **TPR at low FPR.** Carlini *et al.* [2] suggested using TPR at low FPR to measure MIA. A higher TPR in the low FPR region indicates greater precision of the MIA, which also implies that the attack is more reliable.
- **Balance accuracy and AUC.** These two metrics assess the overall performance of MIA. Balanced accuracy measures the attacker's ability to correctly predict true positives and true negatives across all members and non-members. AUC quantifies the area beneath the ROC curve of the MIA results. It offers a comprehensive measure of the attack's discriminative power across various classification thresholds.

4.1.5 PDA-FD Frameworks. To comprehensively evaluate the privacy leakage in the PDA-FD setting, we evaluate three different PDA-FD frameworks in our experiments: FedMD [23], DS-FL [15], and Cronus [3]. Each of these FD frameworks employs a distinct approach to enhance the robustness of the FD algorithms. As mentioned in Section 2.1, they behave differently during the local updates phase and use different aggregation algorithms during the communication phase. In order to optimize the performance of all the PDA-FD frameworks, we should carefully select the number of training epochs in each round. Following the approach suggested by Li *et al.* in FedMD [23], we initially train the local models to convergence on the private datasets before transitioning to the shorter update cycles during the distillation phase. Specifically, in the first round of local updates, each client performs 20 epochs of training. In the subsequent rounds, this is reduced to 5 epochs of training for each client. The knowledge distillation phase consists of 10 epochs of training for each client. To reduce communication cost, we randomly select 5000 data samples from the public dataset during each communication phase for the CIFAR-10, CIFAR10/CIFAR100, Fashion-MNIST and Purchase datasets, aligned with previous PDA-FD studies [23]. The CINIC-10 dataset, given its difficulty level, uses a set of 10000 randomly selected samples to ensure adequate knowledge transfer. Table 3 presents the performance of the three PDA-FD frameworks across various Dirichlet distributions and four distinct datasets.

4.2 Experiment Results of LDIA

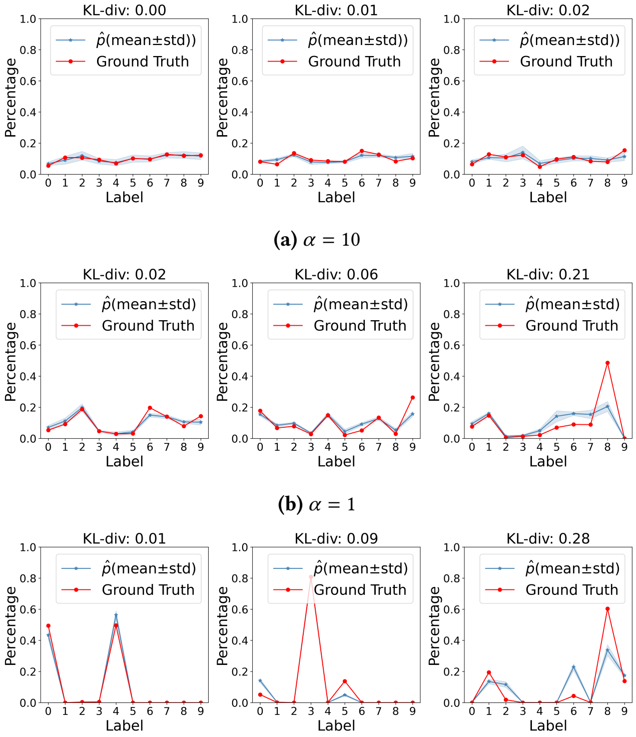
In the LDIA experiments, the PDA-FD server infers the label distribution of all clients' private datasets during the communication phase in each round. To ensure robustness and account for potential fluctuations, we compute the final LDIA result for each client by averaging the server's inferred label distribution over 10 collaborative training rounds. The overall effectiveness of the attack is evaluated by averaging these final results across all clients. To provide a meaningful benchmark for our LDIA method, we establish a baseline comparison, denoted as "Random", following the same approach of previous LDIA research [10, 31]. This baseline employs randomly generated label distributions for each client's private dataset, serving as a lower bound for attack performance. Note that for a given dataset and Dirichlet distribution parameter, the private dataset of each client remains constant across different PDA-FD frameworks. Therefore, within the same dataset and Dirichlet distribution, there is only one set of Random LDIA results.

Table 3: Performance of the PDA-FD Frameworks.

Datasets	Setting	Local accuracy	Federated accuracy		
			FedMD	DS-FL	Cronus
CIFAR10	α=10	54.59%	76.61%	71.31%	70.81%
	α=1	46.06%	75.38%	68.45%	67.90%
	α=0.1	22.75%	60.55%	45.01%	42.24%
CIFAR10 /CIFAR100	α=10	53.24%	72.34%	69.54%	68.42%
	α=1	45.49%	68.41%	65.90%	64.26%
	α=0.1	23.31%	49.89%	43.55%	43.43%
CINIC10	α=10	39.31%	67.72%	64.21%	62.92%
	α=1	33.37%	62.91%	57.02%	56.49%
	α=0.1	20.45%	41.02%	38.48%	37.89%
Fashion -MNIST	α=10	78.80%	88.68%	87.96%	87.62%
	α=1	69.35%	87.98%	85.25%	84.98%
	α=0.1	19.58%	80.62%	52.45%	56.34%
Purchase	α=10	82.56%	94.58%	88.35%	88.62%
	α=1	72.73%	94.01%	86.83%	89.65%
	α=0.1	52.18%	91.80%	72.55%	67.34%

Main Result. Table 4 presents the performance of the proposed LDIA on five different datasets across three PDA-FD frameworks. The results demonstrate the server's capability to launch effective LDIA against clients across these datasets, significantly outperforming the random guess baseline on all three key metrics. For instance, for the DS-FL framework on the CIFAR-10 dataset with α=1, our proposed LDIA achieves an average Chebyshev distance of 0.10, an average mean l1-distance of 0.03, and an average KL-divergence of 0.10 across all clients. In contrast, the random guess baseline yields substantially higher values: 0.20, 0.08, and 0.66 for the respective metrics. This significant improvement underscores the efficacy of our LDIA in accurately inferring clients' label distributions. Figure 6 provides a visual representation of the LDIA results for the DS-FL server on the CIFAR-10 dataset, offering a clearer illustration of the experiment results. We can see from the figure that for the labels whose inferred proportions deviate from the ground truth values, the relative rankings of label frequencies are consistently preserved. This observation highlights the robustness of the proposed LDIA in capturing the essential structure of label distributions.

Different Data Distributions. Our experiments reveal a notable relationship between the effectiveness of the proposed LDIA and the uniformity of clients' label distributions. Specifically, the LDIA demonstrates lower KL-divergence, Chebyshev distance, and mean $l1$-distance as the clients' label distributions become more uniform. This trend is clearly illustrated in our experiments using the CIFAR-10 dataset within the DSFL framework. As α increases, indicating a more uniform label distribution across clients, the LDIA achieves better performance. Conversely, as α decreases, indicating a more skewed distribution, we see an increase in the three key metrics. Nonetheless, the attack remains effective despite the reduced accuracy.



(a) α = 10

(b) α = 1

(c) α = 0.1

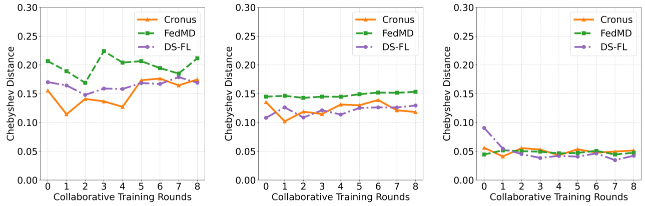
Figure 6: The LDIA performance of the DS-FL server on the CIFAR-10 dataset, under three distinct Dirichlet distributions. The images depict the best (left), median (center), and worst (right) LDIA results across all client models.

Different PDA-FD Frameworks. Our evaluations also reveal significant differences in the vulnerability of various PDA-FD frameworks to LDIA. Compared to FedMD, the server can achieve more effective LDIA on clients within the DS-FL and Cronus frameworks. This can be attributed to the unique training approach employed by FedmD during its first collaborative training round. In FedMD, clients first train their local models on the public dataset before transitioning to their private dataset. This process serves as a form of regularization, thus mitigating overfitting to private datasets and, consequently, reducing vulnerability to LDIA. However, this effect is diminished when private and public datasets differ significantly or when the public dataset is unlabeled. In these cases, FedMD's LDIA vulnerability becomes comparable to that of the other two PDA-FD frameworks, as evidenced by the results in Table 4 for the CIFAR-10/CIFAR-100 datasets. The similarity in LDIA vulnerability arises from the data distribution shift between public and private datasets, causing clients' local models to reduce memorization of the public dataset after converging on their private datasets. As a result, the clients' local models end up overfitting to their private datasets to a similar degree across all frameworks.

Different Collaborative Training Rounds. To evalutate the temporal dynamics of LDIA, we analyze its performance across multiple collaborative training rounds in various PDA-FD frameworks, as illustrated in Figure 7. We aggregate the server's attack results across all clients for each round, to represent the overall LDIA performance over time. The results reveal that the server successfully executes LDIA on clients in every round. Notably, the LDIA performance remains relatively stable as the number of collaborative training rounds increases, showing neither significant improvement nor decline. This consistency can be attributed to the local updates phase preceding communication in each collaborative training round within PDA-FD frameworks. While this phase

Table 4: Performance of the server in conducting LDIA within the different PDA-FD Frameworks.

Datasets	Setting	KL divergence				Chebyshev distance				Mean $l1$-distance			
		FedMD	DS-FL	Cronus	Random	FedMD	DS-FL	Cronus	Random	FedMD	DS-FL	Cronus	Random
CIFAR10	α=10	0.02	0.01	0.01	0.42	0.04	0.03	0.02	0.13	0.01	0.01	0.01	0.05
	α=1	0.17	0.10	0.08	0.66	0.14	0.11	0.10	0.20	0.04	0.03	0.03	0.08
	α=0.1	0.15	0.11	0.07	1.93	0.18	0.16	0.14	0.57	0.04	0.03	0.03	0.15
CIFAR10 /CIFAR100	α=10	0.07	0.05	0.06	0.40	0.07	0.06	0.06	0.12	0.02	0.02	0.02	0.05
	α=1	0.11	0.10	0.10	0.59	0.10	0.11	0.11	0.22	0.03	0.03	0.03	0.08
	α=0.1	0.11	0.10	0.08	1.59	0.15	0.13	0.14	0.51	0.03	0.03	0.03	0.14
CINIC10	α=10	0.01	0.01	0.01	0.64	0.02	0.02	0.04	0.15	0.01	0.01	0.01	0.07
	α=1	0.06	0.05	0.05	0.57	0.11	0.11	0.10	0.21	0.02	0.02	0.02	0.08
	α=0.1	0.09	0.08	0.01	1.99	0.14	0.14	0.04	0.56	0.03	0.03	0.01	0.15
Fashion -MNIST	α=10	0.03	0.02	0.02	0.32	0.04	0.04	0.04	0.12	0.02	0.01	0.01	0.06
	α=1	0.14	0.12	0.12	0.55	0.10	0.09	0.09	0.15	0.04	0.03	0.03	0.06
	α=0.1	0.21	0.05	0.06	1.54	0.20	0.09	0.11	0.45	0.04	0.02	0.02	0.13
Purchase	α=10	0.08	0.03	0.03	0.47	0.06	0.05	0.05	0.13	0.02	0.02	0.02	0.06
	α=1	0.27	0.14	0.15	0.68	0.13	0.10	0.10	0.18	0.06	0.04	0.04	0.09
	α=0.1	0.64	0.14	0.14	2.11	0.32	0.15	0.17	0.52	0.10	0.03	0.03	0.15



(a) α = 0.1 **(b)** α = 1 **(c)** α = 10

Figure 7: Chebyshev distance results of LDIA performed by the PDA-FD server on the CIFAR-10 dataset, shown for each collaborative training round.

enhances knowledge transfer among clients, it simultaneously increases the degree of overfitting of each client's local model to their private data. This dual effect contributes to the observed stability in LDIA performance over multiple rounds. Despite this stability, we recommend using the averaged LDIA results over multiple rounds as the final outcome to enhance the robustness of the results. Such an approach mitigates potential fluctuations and provides a more reliable measure of the server's LDIA capabilities.

4.3 Experiment Results of MIA

In our MIA experiments, we evaluate the effectiveness of the proposed attack against 10 clients during the communication phase across three PDA-FD frameworks. These experiments use three different Dirichlet distributions and four distinct datasets. This comprehensive setup allows for a thorough assessment of MIA vulnerability under various data distribution scenarios and PDA-FD frameworks. Previous FD MIA studies [24, 38] assume that attackers have access to shadow datasets matching the target model's training data distribution. However, we find these assumptions too restrictive and unrepresentative of real-world scenarios. Therefore, our attack methods do not rely on such assumptions, so we do not use these studies as baselines for comparison.

Main Result. We first evaluate co-op LiRA. Given that co-op LiRA is applicable in scenarios where clients' label distributions are similar, we conduct experiments across different datasets using a Dirichlet distribution with α = 10. This parameter setting ensures a more uniform distribution of labels across clients, aligning with co-op LiRA's operational scenario. Table 5 presents the performance of co-op LiRA during the communication phase of the first collaborative training round. Our findings reveal that when the server attacks a specific client, utilizing only the other 9 clients' models as the reference models yields remarkably effective attack results. This observation underscores the high efficiency and practicality of co-op LiRA, demonstrating its capability to achieve effective MIA without the need to train any additional reference models.

We subsequently evaluate the performance of distillation-based LiRA. For each client, we distill 32 reference models. The distillation dataset for each reference model consists of a randomly sampled 80% subset of the public dataset used in the communication phase. This approach ensures a diverse set of reference models. The model architecture of reference model is same as that of the target client model. Table 6 presents the average results of the server's MIA against all clients during the communication phase during the first collaborative training round. The results reveal that the server can launch highly effective MIA against the clients in the non-IID scenarios. Figure 8 presents the results of MIA experiments conducted in the DS-FL framework using the CIFAR-10 dataset, across various Dirichlet distributions. Notably, we observe that even when the private dataset (CIFAR-10) and public dataset (CIFAR-100) have significantly different distributions, the server can still successfully launch MIA against clients by leveraging the distilled reference models from the public dataset. This finding underscores the efficacy of distillation-based LiRA in the PDA-FD frameworks, demonstrating its robustness to dataset disparities between the public and private data.

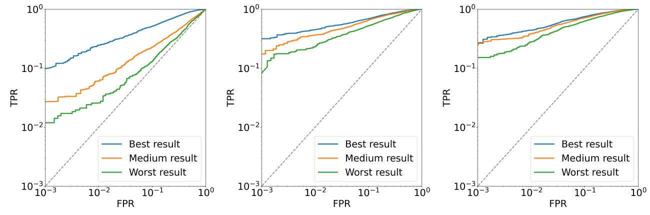
Different Data Distributions. We observe that the effectiveness of the proposed distillation-based LiRA attacks on clients decreases as the clients' label distributions become more imbalanced. This phenomenon can be explained by the fact that local models trained on datasets with highly skewed label distributions tend to produce disproportionately high posterior probabilities for the dominant labels. This bias also affects the non-member samples that come from the same over-represented classes. The core principle of LiRA relies on difficulty calibration, which becomes less

Table 5: Performance of the server in conducting co-op LiRA within the different PDA-FD Frameworks.

Datasets	TPR at 1% FPR			TPR at 0.1% FPR			AUC			Balance Accuracy		
	FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus
CIFAR10	22.15%	21.96%	20.35%	6.67%	6.39%	5.76%	0.819	0.850	0.840	74.07%	77.11%	76.23%
CINIC10	10.97%	11.23%	10.99%	1.78%	1.80%	1.79%	0.794	0.811	0.815	71.55%	73.89%	74.28%
Fashion-MNIST	3.24%	1.80%	1.64%	1.09%	0.31%	0.33%	0.582	0.533	0.531	55.89%	55.38%	54.28%
Purchase	5.85%	4.08%	4.45%	1.67%	0.71%	0.61%	0.616	0.712	0.709	58.41%	66.46%	66.10%

Table 6: Performance of the server in conducting distillation-based LiRA within the different PDA-FD Frameworks.

Datasets	Setting	TPR at 1% FPR			TPR at 0.1% FPR			AUC			Balance Accuracy		
		FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus	FedMD	DS-FL	Cronus
CIFAR10	α=10	20.94%	35.76%	32.69%	10.42%	19.42%	14.40%	0.764	0.902	0.867	69.68%	82.01%	78.51%
	α=1	17.62%	29.28%	23.83%	8.11%	11.29%	5.77%	0.730	0.839	0.804	66.93%	76.20%	72.94%
	α=0.1	9.74%	12.89%	7.20%	2.09%	3.74%	1.13%	0.618	0.680	0.639	58.91%	63.49%	60.64%
CIFAR10 /CIFAR100	α=10	11.20%	34.61%	28.28%	1.48%	10.92%	5.49%	0.804	0.901	0.891	72.74%	81.93%	80.97%
	α=1	9.47%	25.94%	19.32%	0.93%	2.40%	1.95%	0.758	0.844	0.821	68.92%	76.61%	73.68%
	α=0.1	7.79%	11.34%	5.61%	0.74%	0.51%	0.88%	0.627	0.686	0.652	59.38%	63.67%	61.75%
CINIC10	α=10	13.83%	17.32%	15.91%	3.12%	4.15%	3.85%	0.741	0.855	0.834	71.42%	77.57%	75.26%
	α=1	10.96%	13.94%	12.07%	2.37%	3.59%	3.01%	0.704	0.781	0.757	68.93%	70.89%	69.21%
	α=0.1	5.91%	6.81%	6.46%	1.25%	1.94%	1.72%	0.649	0.652	0.661	60.27%	61.18%	62.59%
Fashion -MNIST	α=10	3.07%	1.85%	1.73%	0.91%	0.35%	0.21%	0.588	0.539	0.528	55.94%	59.85%	55.43%
	α=1	3.30%	1.71%	1.62%	0.69%	0.26%	0.25%	0.583	0.536	0.522	56.47%	59.51%	54.31%
	α=0.1	1.88%	1.39%	1.21%	0.54%	0.19%	0.23%	0.538	0.523	0.519	52.71%	52.35%	51.32%
Purchase	α=10	1.94%	5.91%	2.43%	0.77%	1.31%	1.93%	0.539	0.665	0.706	53.34%	62.69%	65.62%
	α=1	1.98%	5.64%	2.69%	0.83%	1.69%	0.68%	0.534	0.654	0.653	53.31%	61.49%	62.24%
	α=0.1	1.41%	5.04%	3.04%	0.42%	1.21%	1.19%	0.507	0.591	0.588	52.24%	57.93%	57.69%



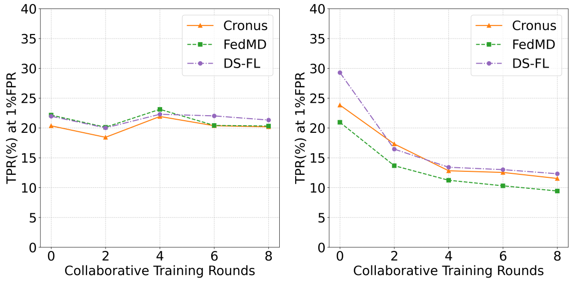
(a) α = 0.1 (b) α = 1 (c) α = 10

Figure 8: Distillation-based LiRA performance of the DS-FL server on the CIFAR-10 dataset, presented as log-scale ROC curves under three distinct Dirichlet distributions.

effective in imbalanced scenarios. As a result of this, the attacker’s capability to discriminate between members and non-members is compromised. This leads to a overall degradation in the performance of distillation-based LiRA on clients with highly imbalanced label distributions.

Different PDA-FD Frameworks. The effectiveness of co-op LiRA remains relatively consistent across FedMD, DS-FL, and Cronus. However, for distillation-based LiRA, the effectiveness of MIAs ranks as follows: DS-FL achieves the best performance, followed by Cronus, with FedMD showing the least effectiveness. In Cronus, clients upload softmax-processed posterior probability vectors rather than raw logits for public data during the communication phase. Compared to logits, the use of posterior probability vectors diminishes the server’s ability to distill reference models that closely mimic the target model’s performance. Consequently, this limitation leads to a reduction in the effectiveness of MIA. In the FedMD framework, clients train on public data before their private datasets during the local updates phase in their first collaborative training round. This process leads to clients training on all the target samples strategically selected by the server, regardless of their membership status. While experiments demonstrate that subsequent training on private datasets reduces clients’ memorization of public data, this initial exposure still impacts the server’s MIA results. However, when the public dataset is unlabeled, clients cannot train on it during the first collaborative training round. In this scenario, the server’s MIA performance on clients remains unaffected.

Different Collaborative Training Rounds. Figure 9 illustrates the performance of the proposed MIAs in different PDA-FD frameworks across multiple collaborative training rounds on the CIFAR-10 dataset. To evaluate each MIA approach in its intended scenario, for co-op LiRA, we employ a Dirichlet distribution parameter α=10. While for distillation-based LiRA, we use α=1. We use the average TPR at 1% FPR of MIA across all clients as the metric to quantify performance. The performance of distillation-based LiRA declines with more collaborative training rounds but eventually stabilizes. We attribute this to the FD process, which gradually reduces the degree of overfitting of local models to their private data. While local updates maintain some private data memorization, the performance gap between local and distilled reference models for private data narrows over time. The performance of co-op LiRA remains relatively stable as the collaborative training rounds increase. We attribute this stability to two key factors. First, the non-target clients in co-op LiRA cannot effectively learn membership information from the server-aggregated logits during the communication phase. Second, while the FD process reduces the overfitting level of local models to their private data, the local updates phase, where each client trains exclusively on its private dataset, maintains a consistent performance gap between local and reference models for their respective private data. This balance between reduced overfitting and continued exclusive training on private data likely contributes to the stability of co-op LiRA's performance across collaborative rounds.



(a) Co-op LiRA (b) Distillation-based LiRA

Figure 9: MIA performance across training rounds.

5 Ablation Study

5.1 Public Dataset Size

The server in PDA-FD can control communication overhead by adjusting the size of the public dataset used in each collaborative training round. We investigate its impact on the performance of LDIA and MIA. As public data does not affect co-op LiRA, our evaluations of MIA mainly focus on distillation-based LiRA. In our experiments, we use the DS-FL framework on the CIFAR-10 dataset with $\alpha = 1$. Table 7 illustrates the degree of label distribution information and membership information leakage from clients when the quantity of the public data samples is set to 5000, 7500, and 10000, respectively. The results indicate that larger public datasets contribute to increased privacy leakage risks for clients. We attribute this trend to two factors. For distillation-based LiRA, a larger public dataset provides a more extensive distillation dataset, enabling the attacker to obtain more robust reference models. In the case of LDIA, a larger public dataset serving as the inference dataset allows the attacker to mitigate the impact of outliers or atypical data, thereby improving attack accuracy.

Table 7: Impact of Public Data Quantity on Label Distribution and Membership Information Leakage in PDA-FD.

Datasets size	MIA (TPR at 1%FPR)	LDIA (KL divergence)
5000	29.28%	0.10
7500	31.84%	0.09
10000	32.01%	0.07

5.2 Number of Epochs in Local Updates Phase

Prior to the communication phase, clients train their local models on their private datasets during the local updates phase. This process enhances the local model's memorization of private data, facilitating knowledge transfer between clients but also potentially increasing privacy leakage. We measure the impact of the number of training epochs in the local updates phase on the leakage of label information and membership information from clients. As shown in Table 8, there is an increase in label distribution and membership information leakage from clients in DS-FL as the number of the local update training rounds increases from 2 to 6 on the CIFAR-10 dataset (α=1).

Table 8: Impact of Number of Training Epochs on Label Distribution and Membership Information Leakage in PDA-FD.

Number of Epochs	MIA (TPR at 1%FPR)	LDIA (KL divergence)
2	8.10%	0.15
4	14.64%	0.10
6	15.43%	0.09

5.3 Number of Reference Models

In LiRA, the attacker can form a more accurate Gaussian distribution by utilizing a larger number of reference models, thereby enhancing the precision of determining whether a target sample belongs to the target model's training data. We evaluate the performance of distillation-based LiRA with varying numbers of reference models. Figure 10 shows results from experiments using the Cronus framework on CIFAR-10 with $\alpha = 0.1$. The data reveals that the performance of the distillation-based LiRA's improves as the number of distilled reference models increases.

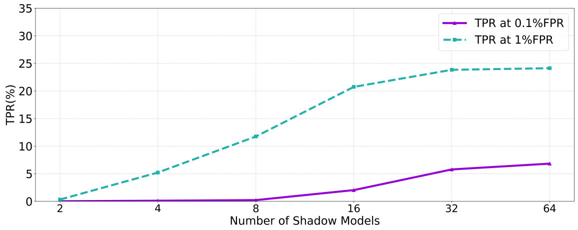


Figure 10: The performance of distillation-based LiRA vs. number of the distilled reference model.

5.4 Resilience Against DP-SGD

To evaluate the robustness of our proposed LDIA and MIA methods, we assess their effectiveness when the target client employs DP-SGD[1] during the local updates phase. DP-SGD is a state-of-the-art privacy-preserving model training technique. Our experimental setup includes 10 clients participating in DS-FL training on the CIFAR-10 dataset ($\alpha = 10$). We conduct LDIA and Co-op LiRA attacks against the clients during the second round of training. In DP-SGD, it introduces noise to gradients during training, governed by three key parameters. The clipping bound (C) limits the influence of individual data points on model parameters. The noise multiplier (σ) determines the amount of noise added to gradients. The privacy budget (ε) balances privacy guarantees and model utility, with smaller values providing stronger privacy at the cost of potentially noisier updates. In our experiments, we set C to be 10 and vary σ to adjust ε. This setup allows us to evaluate our proposed attack under different privacy protection levels.

Table 9: Performance of MIA and LDIA against DP-SGD for DS-FL trained on CIFAR-10.

σ	ε	Average acc	LDIA(KL divergence)	MIA(TPR at 1%FPR)
0	∞	59.09%	0.03	15.76%
0.1	>10000	48.68%	0.07	2.86%
0.3	>5000	41.29%	0.08	2.11%
0.5	>2000	28.53%	0.09	1.54%
1.0	231	21.34%	0.10	1.29%

5.5 Resilience Against Evasive Clients

To proactively protect their privacy, cautious clients may choose to, in each communication round, avoid sending to the server the logits of some samples in the public dataset, particularly the ones that are also in its training dataset. To counter such defense, we propose two countermeasures as follows: (1) In co-op LiRA, a shadow target model can be distilled using the logits of the samples in the public dataset provided by the target model, and then obtain the logits of the target sample from this shadow target model as an approximation to the one from the target clients' model. The intuition is that although knowledge distillation reduces the distilled student model's membership information of the teacher model [16], it still preserves statistically significant enough membership information for a percentage of the members in teacher's training data, thus allowing some success in the MIA attack to the teacher model. (2) We can also leverage a technique called indirect queries [25, 40], which is to first obtain logits of samples in the target sample's neighborhood from the target model and subsequently perform MIA using information encoded in these neighborhood logits. Neighbor samples are generated by adding noises to the target sample.

We conduct experiments to evaluate the effectiveness, and the experiments are on the CIFAR-10 dataset with a Dirichlet distribution parameter α=10, with co-op LiRA as the MIA method. Equipped with the first countermeasure, the attack achieves a TPR of 4.53% at 1% FPR. Implementing a simplified version of the second countermeasure gives the attack a TPR of 4.23% at 1% FPR. Note that in implementing countermeasure two, we add random Gaussian noise to the target samples to generate neighbor samples, with the noise clipped to the [-0.7, 0.7] range. Studies in [25, 40] implement more advanced schemes to learn from the neighbor logits, leading to better attacks. We leave studying such schemes as future work.

6 Privacy Risk in Federated Distillation

While FL is designed to protect clients' private data, recent research [10, 24, 28, 38, 43] reveals significant privacy risks in these frameworks. In FL, Gu *et al.* [10] demonstrated that server-side LDIA could achieve a KL-divergence of 0.01 between the inferred and the ground truth label distributions on CIFAR-10. Nasr *et al.* [28] showed that server or client-side MIA could reach accuracies of 92.1% and 76.3%, respectively, on CIFAR-100.

FD frameworks transfer distilled knowledge between participants instead of informative model parameters and gradients. This mechanism generally provides more privacy protection for each client's data than traditional FL frameworks (FedAVG, FedSGD, etc.). However, through the lens of LDIA and MIA, we observe that although privacy leakage risk in FD appears less severe than in FL, significant risks remain, as state-of-the-art privacy attacks can still achieve non-trivial success rates based on the results in the literature and our experiments. Our work is the first to propose a LDIA method targeting the FD frameworks, and we achieve a KL divergence of 0.02 between the inferred and the ground-truth label distributions on CIFAR-10. This attack is less successful than in the traditional FL frameworks, but label distribution leakage has been demonstrated. Targeting the PDA-FD frameworks, Liu *et al.* [24] proposed a client-side MIA method attaining 67.0% balanced accuracy on CIFAR-100. Yang *et al.* [43] also demonstrated a client-side MIA method that achieved an up to 75% balanced accuracy on CIFAR-100. Similarly, our MIA methods (co-op LiRA and distillation-based LiRA) demonstrate considerable server-side MIA effectiveness in achieving a TPR of up to 35.76% at a 1% FPR on CIFAR-10. In addition, effective MIA methods are reported to target other FD frameworks. For example, Wang *et al.* [38] reported that their MIA attack achieved 67.06% and 79.07% accuracy on FedGen [45] and FedDistill [18] respectively, on CIFAR-10. One of the objectives of our study is to motivate future research on privacy risks in various FD frameworks and, more broadly, FL frameworks.

7 Related Work

FL has emerged as a crucial learning scheme for distributed training that aims to preserve user data privacy. However, research has uncovered various privacy vulnerabilities in FL, particularly in the form of MIA and LDIA. This section discusses relevant works that highlight these threats in both FL and FD settings, and contextualize our research within this landscape.

MIA and LDIA. Shokri *et al.* [34] pioneered MIA research by demonstrating how model output confidence scores could reveal training data membership. Nasr *et al.* [28] extended this to FL, showing how both passive and active adversaries could exploit gradients and model updates. LDIA represents another significant privacy threat in FL. Gu *et al.* [10] introduced LDIA as a new attack vector where adversaries infer label distributions from model updates. Wainakh *et al.* [37] further explored user-level label leakage through gradient-based attacks in FL. Recent works have exposed the vulnerability of FD to inference attacks. Yang *et al.* [43] proposed FD-Leaks for performing MIA in FD settings through logit analysis. Liu *et al.* [24] and Wang *et al.* [38] enhanced MIA using shadow models via respective approaches MIA-FedDL and GradDiff, though their assumptions were limited to homogeneous environments.

Defenses and Countermeasures. DPSGD [1] can be employed during the training phase to mitigate against privacy attacks to the client model. Additionally, specialized MIA defense methods such as SELENA [36], HAMP [5] and DMP[33] can be integrated into the training process. Several studies have proposed enhanced FD frameworks with improved privacy protection mechanisms to reduce client privacy leakage. Zhu *et al.* [45] investigated data-free knowledge distillation for heterogeneous federated learning. They presented an approach that reduces the need for public datasets. Chen *et al.* [4] proposed FedHKD, where clients share hyper-knowledge based on data representations from local datasets for federated distillation without requiring public datasets or models.

8 Conclusion

In this paper, we examine the privacy risk of using public datasets as the knowledge transfer medium in FD through the lens of label distribution information and membership information leakage, measured by attack success rates. We evaluate three public-dataset-assisted FD frameworks (FedMD, DS-FL, and Cronus) using our proposed LDIA method and two MIA methods: Co-op LiRA and Distillation-based LiRA. Our LDIA method performs considerably well measured in KL divergence (an average KL divergence of 0.10 in one setup), demonstrating a non-trivial risk level of label distribution information leakage. Co-op LiRA and Distillation-based LiRA, two MIA attacks for FD, achieve a state-of-the-art success rate evident by relatively high TPRs at low FPRs (up to 34.61% TPR at 1% FPR), indicating troublesome membership leakage risk. These findings underscore the privacy vulnerabilities that persist in PDA-FD frameworks, highlighting the need for enhanced privacy-preserving mechanisms in FD environments.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 308–318.

[2] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1897–1914.

[3] Hongyan Chang, Virat Shejwalkar, Reza Shokri, and Amir Houmansadr. 2019. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. *arXiv preprint arXiv:1912.11279* (2019).

[4] Huancheng Chen, Haris Vikalo, et al. 2023. The best of both worlds: Accurate global and personalized models through federated learning with data-free hyper-knowledge distillation. *arXiv preprint arXiv:2301.08968* (2023).

[5] Zitao Chen and Karthik Pattabiraman. 2023. Overconfidence is a dangerous thing: Mitigating membership inference attacks by enforcing less confident prediction. *arXiv preprint arXiv:2307.01610* (2023).

[6] Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. 2018. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505* (2018).

[7] Ilias Diakonikolas, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. 2017. Being robust (in high dimensions) can be practical. In *International Conference on Machine Learning*. PMLR, 999–1008.

[8] DMDave, Todd B, and Will Cukierski. 2014. Acquire Valued Shoppers Challenge. https://kaggle.com/competitions/acquire-valued-shoppers-challenge. Kaggle.

[9] Kang Fu, Dawei Cheng, Yi Tu, and Liqing Zhang. 2016. Credit card fraud detection using convolutional neural networks. In *Neural Information Processing: 23rd International Conference, ICONIP 2016, Kyoto, Japan, October 16–21, 2016, Proceedings, Part III 23*. Springer, 483–490.

[10] Yuhao Gu and Yuebin Bai. 2023. LDIA: Label distribution inference attack against federated learning in edge computing. *Journal of Information Security and Applications* 74 (2023), 103475.

[11] Mohammad Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Pal, Pierre-Marc Jodoin, and Hugo Larochelle. 2017. Brain tumor segmentation with deep neural networks. *Medical image analysis* 35 (2017), 18–31.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[13] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[14] Yue Huang, Lanju Kong, Qingzhong Li, and Baochen Zhang. 2023. Decentralized Federated Learning Via Mutual Knowledge Distillation. In *2023 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 342–347.

[15] Sohei Itahara, Takayuki Nishio, Yusuke Koda, Masahiro Morikura, and Koji Yamamoto. 2021. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data. *IEEE Transactions on Mobile Computing* 22, 1 (2021), 191–205.

[16] Matthew Jagielski, Milad Nasr, Katherine Lee, Christopher A Choquette-Choo, Nicholas Carlini, and Florian Tramer. 2024. Students parrot their teachers: Membership inference on model distillation. *Advances in Neural Information Processing Systems* 36 (2024).

[17] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. 2018. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. *arXiv preprint arXiv:1811.11479* (2018).

[18] Donglin Jiang, Chen Shan, and Zhihui Zhang. 2020. Federated learning algorithm based on knowledge distillation. In *2020 International conference on artificial intelligence and computer engineering (ICAICE)*. IEEE, 163–167.

[19] Yangfan Jiang, Xinjian Luo, Yuncheng Wu, Xiaokui Xiao, and Beng Chin Ooi. 2024. Protecting Label Distribution in Cross-Silo Federated Learning. In *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 113–113.

[20] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[21] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.

[22] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[23] Daliang Li and Junpu Wang. 2019. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581* (2019).

[24] Siqi Liu and Fang Dong. 2023. MIA-FedDL: A Membership Inference Attack against Federated Distillation Learning. In *2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*. IEEE, 1148–1153.

[25] Yunhui Long, Lei Wang, Diyue Bu, Vincent Bindschaedler, Xiaofeng Wang, Haixu Tang, Carl A Gunter, and Kai Chen. 2020. A pragmatic approach to membership inferences on machine learning models. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 521–534.

[26] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.

[27] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 691–706.

[28] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 739–753.

[29] Andrew Paverd, Andrew Martin, and Ian Brown. 2014. Modelling and automatically analysing privacy properties for honest-but-curious adversaries. *Tech. Rep* (2014).

[30] Raksha Ramakrishna and György Dán. 2022. Inferring Class-Label Distribution in Federated Learning. In *Proceedings of the 15th ACM Workshop on Artificial Intelligence and Security*. 45–56.

[31] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. 2020. {Updates-Leak}: Data set inference and reconstruction attacks in online learning. In *29th USENIX security symposium (USENIX Security 20)*. 1291–1308.

[32] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4510–4520.

[33] Virat Shejwalkar and Amir Houmansadr. 2021. Membership privacy for machine learning models through knowledge transfer. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 9549–9557.

[34] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*. IEEE, 3–18.

[35] Dianbo Sui, Yubo Chen, Jun Zhao, Yantao Jia, Yuantao Xie, and Weijian Sun. 2020. Feded: Federated learning via ensemble distillation for medical relation extraction. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*. 2118–2128.

[36] Xinyu Tang, Saeed Mahloujifar, Liwei Song, Virat Shejwalkar, Milad Nasr, Amir Houmansadr, and Prateek Mittal. 2022. Mitigating membership inference attacks by {Self-Distillation} through a novel ensemble architecture. In *31st USENIX Security Symposium (USENIX Security 22)*. 1433–1450.

[37] Aidmar Wainakh, Fabrizio Ventola, Till Müßig, Jens Keim, Carlos Garcia Cordero, Ephraim Zimmer, Tim Grube, Kristian Kersting, and Max Mühlhäuser. 2021. User-level label leakage from gradients in federated learning. *arXiv preprint arXiv:2105.09369* (2021).

[38] Xiaodong Wang, Longfei Wu, and Zhitao Guan. 2024. GradDiff: Gradient-based membership inference attacks against federated distillation with differential comparison. *Information Sciences* 658 (2024), 120068.

[39] Lauren Watson, Chuan Guo, Graham Cormode, and Alex Sablayrolles. 2021. On the importance of difficulty calibration in membership inference attacks. *arXiv preprint arXiv:2111.08440* (2021).

[40] Yuxin Wen, Arpit Bansal, Hamid Kazemi, Eitan Borgnia, Micah Goldblum, Jonas Geiping, and Tom Goldstein. 2022. Canary in a coalmine: Better membership inference with ensembled adversarial queries. *arXiv preprint arXiv:2210.10750* (2022).

[41] Chuhan Wu, Fangzhao Wu, Lingjuan Lyu, Yongfeng Huang, and Xing Xie. 2022. Communication-efficient federated learning via knowledge distillation. *Nature communications* 13, 1 (2022), 2032.

[42] H Xiao. 2017. Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747* (2017).

[43] Zilu Yang, Yanchao Zhao, and Jiale Zhang. 2022. Fd-leaks: Membership inference attacks against federated distillation learning. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. Springer, 364–378.

[44] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*. IEEE, 268–282.

[45] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. 2021. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*. PMLR, 12878–12889.