Where Does This Data Come From? Enhanced Source Inference Attacks in Federated Learning

Haiyang Chen[1], **Xiaolong Xu**[1*], **Xiang Zhu**[2], **Xiaokang Zhou**[3], **Fei Dai**[4], **Yansong Gao**[5], **Xiao Chen**[6], **Shuo Wang**[7], **Hongsheng Hu**[6]
[1]Nanjing University of Information Science and Technology
[2]National University of Defense Technology
[3]Kansai University
[4]Southwest Forestry University
[5]The University of Western Australia
[6]University of Newcastle
[7]Shanghai Jiao Tong University
{oceanchen66, njuxlxu, flydai.cn}@gmail.com, zhuxiang@nudt.edu.cn, zhou@kansai-u.ac.jp, garrison.gao@uwa.edu.au, shellchen1003@hotmail.com, wangshuosj@sjtu.edu.cn, hongsheng.hu@newcastle.edu.au

Abstract

Federated learning (FL) enables collaborative model training without exposing raw data, offering a privacy-aware alternative to centralized learning. However, FL remains vulnerable to various privacy attacks that exploit shared model updates, including membership inference, property inference, and gradient inversion. Source inference attacks further threaten FL by identifying which client contributed a specific training sample, posing severe risks to user and institutional privacy. Existing source inference attacks mainly assume passive adversaries and overlook more realistic scenarios where the server actively manipulates the training process. In this paper, we present an enhanced source inference attack that demonstrates how a malicious server can amplify behavioral differences between clients to more accurately infer data origin. Our approach introduces active training manipulation and data augmentation to expose client-specific patterns. Experimental results across five representative FL algorithms and multiple datasets show that our method significantly outperforms prior passive attacks. These findings reveal a deeper level of privacy vulnerability in FL and call for stronger defense mechanisms under active threat models.

1 Introduction

Federated learning (FL) is a distributed machine learning paradigm that enables multiple clients to collaboratively train a shared model without exposing their raw data [McMahan *et al.*, 2017]. This approach is motivated by growing concerns over data privacy, regulatory constraints, and the need to leverage decentralized data sources. By keeping data local and only exchanging model updates, FL significantly reduces privacy risks and communication overhead [Li *et al.*, 2020b; Li *et al.*, 2021b]. It has gained widespread adoption in various domains, including mobile device personalization, healthcare, finance, and smart manufacturing. The decentralized and privacy-aware nature of FL makes it a promising solution for learning in sensitive and data-siloed environments [Li *et al.*, 2020a; Khodak *et al.*, 2021].

Despite its privacy-preserving design, federated learning remains vulnerable to various privacy attacks that exploit shared model updates [Lyu *et al.*, 2022; Geiping *et al.*, 2020; Wang *et al.*, 2022; Jin *et al.*, 2025; Zhang and Xia, 2024; Zhou *et al.*, 2022]. These attacks can infer sensitive information about individual participants or their local datasets, undermining the core privacy guarantees of FL. A key reason for such vulnerabilities is that gradient updates and model parameters often leak unintended data patterns, especially when models are overparameterized or updates are sparsely aggregated. One prominent type of privacy attack is the membership inference attack, where an adversary aims to determine whether a particular data sample was part of a client's local training set [Shokri *et al.*, 2017]. In the context of federated learning, such attacks can be launched by malicious servers or clients, leveraging access to intermediate model states or updates to infer the privacy of training data [Nasr *et al.*, 2019].

In federated learning, source inference attacks [Hu *et al.*, 2021; Hu *et al.*, 2023; Li *et al.*, 2025] go a step beyond membership inference by aiming to identify which client a specific training sample originated from. By correlating model updates with client-specific data patterns, adversaries can deanonymize participants in the federation, which poses serious privacy risks. For example, in a federated learning system for disease prediction, revealing that a particular hospital contributed data for a rare condition could expose the hospital's patient demographics or even individual patient identi-

*Corresponding Author

ties. Such attacks compromise not only data privacy but also institutional confidentiality and trust in collaborative learning frameworks [Devakumar *et al.*, 2020; Xu *et al.*, 2020]. To launch such attacks, existing attacks [Hu *et al.*, 2021; Hu *et al.*, 2023] leverage the observation that the local model from the source client will produce higher prediction confidences in predicting the target sample than the local models from other clients, because the local model was trained to minimize prediction loss on the target sample directly. Thus, by identifying which client's model has the smallest prediction loss on the target sample, the server can infer its source private information.

Although existing research on source inference attacks [Hu *et al.*, 2021; Hu *et al.*, 2023] has demonstrated their feasibility, most approaches focus on passive adversaries that observe model updates without influencing the training process. However, in practical federated learning deployments, the server, often assumed to be semi-honest or even malicious, can actively manipulate training dynamics to enhance its attack capabilities [Nasr *et al.*, 2019; Kumar *et al.*, 2023; Xie *et al.*, 2022]. In fact, active server-side attacks are common in many federated learning threat models, including backdoor attacks [Bagdasaryan *et al.*, 2020], poisoning attacks [Tolpegin *et al.*, 2020], and gradient inversion attacks [Wei *et al.*, 2025]. However, the impact of actively launched source inference attacks by the server remains underexplored. Investigating such active strategies can reveal more severe and realistic privacy vulnerabilities that better reflect real-world risks. This highlights a critical research gap in understanding the full extent of source leakage in federated learning systems.

In this paper, we propose an enhanced source inference attack in federated learning, demonstrating that a malicious server can actively amplify the behavioral differences between the source client and other clients to more accurately infer the origin of a specific training sample. Given a target sample, the server leverages gradient ascent to intentionally reduce the global model's prediction confidence on that sample during training. As a result, the source client's local model trained directly on the target sample to minimize its loss continues to exhibit high prediction confidence. In contrast, other clients, which do not possess the target sample, adapt their models based on the manipulated global model and consequently produce much lower confidence scores. To further enlarge this discrepancy, we apply data augmentation techniques to generate multiple variants of the target sample, which serve two purposes: guiding the global model to consistently suppress prediction confidence across diverse inputs and helping reveal more stable and distinguishable responses from the source client. The motivation behind using data augmentation is to enhance the model's sensitivity to the presence or absence of the target sample in a client's training data, thereby increasing the effectiveness of the source inference.

Contributions. The main contributions are as follows:

- We propose an enhanced source inference attack in federated learning, demonstrating that a malicious server can actively infer the source of data samples used during federated training. This active attack model reveals a more severe privacy vulnerability in federated learning systems, where the server, often assumed to be semi-trusted, can deliberately manipulate training dynamics to compromise client-level data anonymity.
- Our attack method introduces a novel combination of gradient ascent and data augmentation to deliberately amplify behavioral differences between the source client and non-source clients, significantly outperforming existing passive source inference attacks.
- We conduct extensive experiments across five representative federated learning frameworks using diverse datasets and settings, and the results consistently validate the effectiveness and superiority of our method compared to baseline attacks, with our experimental findings demonstrating superior results.

2 Related Work

Federated Learning. FL is a decentralized machine learning paradigm that enables multiple clients to collaboratively train a global model without sharing their raw data [McMahan *et al.*, 2017]. This design is motivated by increasing concerns over data privacy, security regulations, and the need to utilize sensitive data distributed across different sources [Li *et al.*, 2021a]. In a typical FL process, each client trains a local model on its private dataset and sends model updates, such as gradients or model parameters, to a central server. The server then aggregates these updates to produce a new global model, which is broadcast back to the clients for the next training round. This process is repeated iteratively until the global model converges. Communication is typically structured in synchronous rounds, and only model updates, not raw data, are exchanged. Various federated learning algorithms have been proposed to improve convergence and robustness, such as FedAvg [McMahan *et al.*, 2017], which performs weighted averaging of client updates; FedProx [Li *et al.*, 2020b], which addresses data heterogeneity by adding a proximal term to the loss function; and FedNova [Wang *et al.*, 2020], which normalizes updates to handle varying local training epochs. Other frameworks like SCAFFOLD [Karimireddy *et al.*, 2020] and FedDyn [Durmus *et al.*, 2021] aim to mitigate client drift and improve stability in non-IID settings. Despite its privacy-preserving intent, FL remains vulnerable to various attacks due to the exposure of model updates during training.

Privacy Attacks. Although federated learning is designed to protect raw data by keeping it on local devices, it remains vulnerable to privacy attacks due to the exposure of model updates during training [Lyu *et al.*, 2020; Feng *et al.*, 2024]. These updates can inadvertently leak sensitive information about the underlying data, especially when models are overparameterized or data is non-IID. Membership inference attacks aim to determine whether a specific data sample was part of a client's training set, potentially exposing participation in sensitive activities [Nasr *et al.*, 2019]. Property inference attacks go further by inferring statistical or semantic properties of a client's local dataset that are unrelated to the main learning task, such as demographics or data distribution [Melis *et al.*, 2019]. Gradient inversion attacks

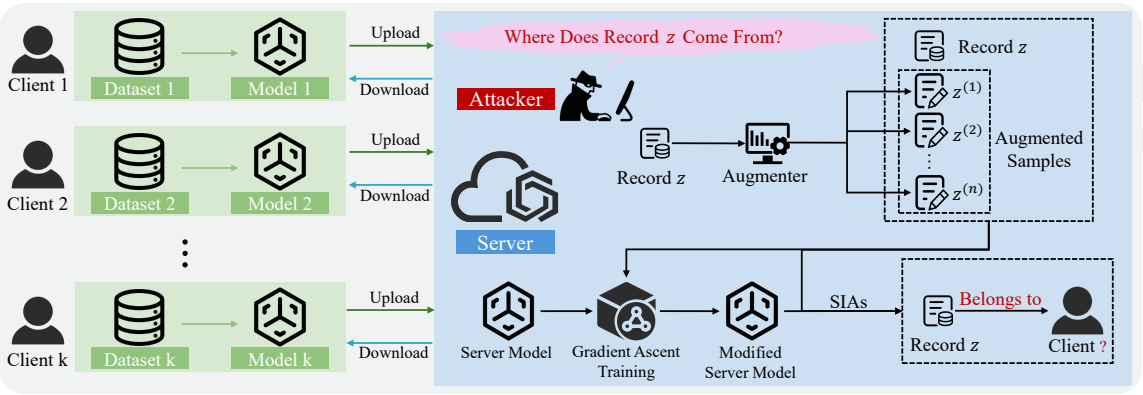

Figure 1: Overview of the proposed source inference attacks in federated learning. Given a data sample, the malicious server first uses data augmentation techniques to create different augmented versions of the sample. Then, the server updates the global model with these samples using gradient ascent. Last, the server sends back the updated global model to each of the clients.

attempt to reconstruct original input data from shared gradients, often recovering surprisingly accurate visual or textual representations [Zhu *et al.*, 2019; Zhao *et al.*, 2020; Geiping *et al.*, 2020]. These attacks exploit the correlation between gradient updates and the input samples that generated them.

Source inference attacks (SIAs) [Hu *et al.*, 2021; Hu *et al.*, 2023; Li *et al.*, 2025] build on the foundation of membership inference attacks by not only identifying whether a sample was used in training but also determining which client it came from. While membership inference focuses on data presence, source inference shifts attention to data origin, revealing a new dimension of privacy risk in federated learning. This source-level perspective has the potential to connect with other privacy attacks, as knowing the source client can enhance the precision or impact of attacks like property inference or gradient inversion. In fact, a stronger source inference signal may serve as an indicator or enabler of stronger attacks in other categories. Therefore, studying source inference attacks provides a broader understanding of privacy vulnerabilities and their interdependencies in FL systems.

3 Threat Model and Our Attack

3.1 Threat Model

Target FL Systems. We focus on horizontal FL frameworks, where each client holds a complete set of features for different data instances and collaboratively trains a global model. In this setting, the training task is shared across clients with similar feature spaces but different data samples. This aligns with most existing works [Hu *et al.*, 2021; Hu *et al.*, 2023; Li *et al.*, 2025] on source inference attacks and is widely adopted in real-world FL deployments. Although our work centers on horizontal FL, vertical FL, where clients hold different features of the same data instances, is also a compelling direction. In such cases, identifying which client owns a particular feature of a sample would present a different form of source inference, which we leave for future exploration.

Attack Goal. The goal of a source inference attacker in FL is to determine which client owns a specific training sample. This threat is particularly concerning in applications where the origin of the data carries sensitive or identifying information. For example, in an FL-based image classification task, revealing which client holds a sensitive image can directly compromise user privacy, especially when the image content relates to personal identity, health, or location [Melis *et al.*, 2019]. Such attacks not only expose individual users but can also breach organizational confidentiality in domains like healthcare or finance.

Attack Knowledge. We assume the central server acts as the adversary conducting SIAs in FL. While performing its standard role in coordinating the FL protocol and aggregating updates to train the global model, the server simultaneously attempts to infer the source client of specific training data based on legitimate communications from the clients. These communications may include gradient updates [McMahan *et al.*, 2017], model parameters [McMahan *et al.*, 2017], or prediction outputs on unlabeled data [Li and Wang, 2019]. The server is also allowed to actively manipulate the training process, such as altering model updates or objectives, to amplify client, specific signals, provided that such manipulations do not significantly degrade the final model utility.

In the SIA setting, the attacker is given a specific data instance—referred to as the target record—which is assumed to have already been identified as part of the training set through a prior membership inference attack. Importantly, this work does not focus on how the attacker obtains the target record, which may be achieved through methods like gradient inversion [Zhu *et al.*, 2019; Geiping *et al.*, 2020]; instead, we investigate whether and how the source identity of such a record can be inferred. An SIA is considered successful if the server can correctly identify the client that contributed the target record to the training process.

3.2 Our Attack

Figure 1 outlines the pipeline of our proposed Enhanced Source Inference Attacks (ESIAs). Consider a target record, z, that is utilized in the federated training process. In each training iteration t, the server initially applies data augmentation to z, yielding an augmented sample set denoted as

$Z_{\text{aug}} = \{z^{(1)}, z^{(2)}, \ldots, z^{(n)}\}$. After aggregating the global model (potentially from client updates in a preceding step), the server performs gradient ascent specifically on the target record z and its augmented counterparts Z_{aug}. This operation is designed to amplify the discrepancies in gradients attributable to z across different clients, while also intentionally degrading the aggregated model's performance on z and Z_{aug}. Following this, the server disseminates the updated gradients (or the modified model reflecting these changes) to the clients. These clients, in turn, update their local models based on the received information and subsequently upload their local model updates back to the server. Finally, the server conducts a source inference analysis on z and Z_{aug}, potentially leveraging the distinct client model updates. The weighted results derived from this inference process are then used to determine the source client of the record z. The detailed description of the data augmentation and gradient ascent implementation is as follows.

Data Augmentation. Traditional SIAs identify the owner of a target record z by evaluating the loss of each client's model on z, hypothesizing that the client with the smallest loss is the owner. However, this approach is vulnerable to noise and incidental factors. For example, a client without z in their training data might still achieve a low loss on z due to similarities with other records, leading to misidentification. To address this limitation, we enhance SIAs by incorporating data augmentation, leveraging the insight that a client genuinely possessing z will exhibit consistently low losses not only on z but also on its augmented variants.

We employ standard data augmentation techniques, such as rotation, flipping, cropping, scaling, and noise addition, to generate diverse yet faithful variations of z. These transformations produce an augmented dataset $Z_{\text{aug}} = \{z^{(1)}, z^{(2)}, \ldots, z^{(n)}\}$, where each $z^{(i)}$ is a perturbed version of z. To ensure relevance, we control the magnitude of these perturbations, balancing diversity (to capture a range of transformations) and fidelity (to preserve similarity to z). This approach enhances the robustness of SIAs by evaluating client models across z and Z_{aug}, reducing the likelihood of false positives.

Building on prior work [Hu *et al.*, 2023], we formalize the SIA framework with the following definitions:

- $Q(\theta_k, z) = \mathbb{E}_\tau\left[\sigma\left(f(z, \theta_k, p_\tau) + \mu_\beta\right)\right]$: Probability that z belongs to client k with model θ_k.
- $L_{p_\tau}(z) = -\alpha \log\left(\int e^{-\frac{1}{\alpha} l(t,z)} p_\tau(t)\, dt\right)$: Expected loss of a general model not trained on z, where $p_\tau(t)$ is the posterior distribution of model parameters excluding z.
- $l(\theta_k, z)$: Loss of client k's model θ_k on z (e.g., cross-entropy loss).
- $f(z, \theta_k, p_\tau) = \frac{1}{\alpha}\left(L_{p_\tau}(z) - l(\theta_k, z)\right)$: Inference score measuring the loss difference, where a higher value suggests z was in client k's local training dataset.

Here, $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid function, μ_β is a bias term reflecting prior ownership probability, and α is a temperature parameter controlling model stochasticity.

With data augmentation, we replace $l(\theta_k, z)$ with an averaged loss over z and its augmentations:

$$l_{\text{aug}}(\theta_k, z) = \frac{1}{n+1}(l(\theta_k, z) + \sum_{i=1}^{n} l(\theta_k, z^{(i)})), \quad (1)$$

where $z^(0) = z$ is implicitly included by defining the average over $n+1$ terms (the original record plus n augmentations). Similarly, the expected loss becomes:

$$L_{\text{aug},p_\tau}(z) = \frac{1}{n+1}(L_{p_\tau}(z) + \sum_{i=1}^{n} L_{p_\tau}(z^{(i)})). \quad (2)$$

The refined inference score is then:

$$f_{\text{aug}}(z, \theta_k, p_\tau) = \frac{1}{\alpha}\left(L_{\text{aug},p_\tau}(z) - l_{\text{aug}}(\theta_k, z)\right). \quad (3)$$

This averaging reduces variance in loss estimates, stabilizing $Q(\theta_k, z)$ and improving the reliability of source inference. By evaluating performance across z and Z_{aug}, the attack more accurately identifies the source client.

Gradient Ascent. In standard federated learning, gradient descent optimizes model parameters by minimizing a loss function, typically expressed as:

$$\theta \leftarrow \theta - \eta \nabla_\theta \ell(\theta), \quad (4)$$

where θ denotes the model parameters, η is the learning rate, and $\nabla_\theta \ell(\theta)$ is the gradient of the loss ℓ. This process updates model parameters to minimize prediction errors.

For ESIAs, we introduce gradient ascent on the global model to amplify differences between clients with and without the target sample z. Gradient ascent updates parameters to *increase* the loss on specific data, formulated as:

$$\theta \leftarrow \theta + \eta \nabla_\theta \ell(\theta; z), \quad (5)$$

where $\ell(\theta; z)$ is the loss on z. After aggregating client updates via FedAvg, the server applies gradient ascent on $theta_t$ using both z and its augmented set Z_{aug}:

$$\theta_t \leftarrow \theta_t + \eta \nabla_\theta \ell(\theta_t; z) + \eta' \cdot \frac{1}{|Z_{\text{aug}}|} \sum_{z' \in Z_{\text{aug}}} \nabla_\theta \ell(\theta_t; z'). \quad (6)$$

This adjustment deliberately degrades the global model's performance on z and its augmentations. The intent is to create a contrast: clients lacking z in their training data receive a global model ill-suited to z, while the client possessing z can leverage local training to recover performance on z and Z_{aug}. In subsequent rounds, this client's local model (θ_k^t) exhibits a lower average loss on z and Z_{aug}, enhancing the attack's ability to identify them as the source.

3.3 ESIAs in FL Frameworks

We investigate ESIAs in five representative FL frameworks to show the broader source privacy vulnerability in FL. Specifically, we investigate SIAs in FedSGD [McMahan *et al.*, 2017], FedAvg [McMahan *et al.*, 2017], FedMD [Li and Wang, 2019], FedPer [Arivazhagan *et al.*, 2019] and FedProx [Li *et al.*, 2020b] where local clients upload gradients, model parameters, or predictions on an unlabeled dataset to

Algorithm 1 Enhanced SIAs in FedAvg

1: **Server executes:**
2: Initialize global model weights θ_0
3: From target record z, generate augmented samples $Z_{\text{aug}} = \{z^{(1)}, z^{(2)}, \ldots, z^{(n)}\}$
4: **for** each round $t = 1$ to T **do**
5: **for** each client $k \in \{1, 2, \ldots, K\}$ **do**
6: $\theta_k^t \leftarrow \text{ClientUpdate}(\theta_{t-1})$
7: **end for**
8: $\theta_t \leftarrow \sum_{k=1}^{K} \frac{n^{(k)}}{n} \theta_k^t$
9: $\theta_t \leftarrow \theta_t + \eta \cdot \nabla_\theta \ell(\theta_t; z) + \eta' \cdot \frac{1}{|Z_{\text{aug}}|} \sum_{z' \in Z_{\text{aug}}} \nabla_\theta \ell(\theta_t; z')$
10: **for** each client $k \in \{1, 2, \ldots, K\}$ **do**
11: Compute $\bar{\ell}_k^t = \ell(\theta_k^t; z) + \frac{1}{|Z_{\text{aug}}|} \sum_{z' \in Z_{\text{aug}}} \ell(\theta_k^t; z')$
12: **end for**
13: $i \leftarrow \arg\min_k \bar{\ell}_k^T$
14: **end for**
15: **return** θ_T, i
16: **ClientUpdate(θ):**
17: $B \leftarrow$ Split local data D_k into batches of size B
18: **for** each local epoch $i = 1$ to E **do**
19: **for** each batch $b \in B$ **do**
20: $\theta \leftarrow \theta - \eta \nabla_\theta \ell(b; \theta)$
21: **end for**
22: **end for**
23: **return** θ

the server. For clarity, we demonstrate the ESIAs algorithm by taking the classic FedAvg framework, but the attack principle applies to other FL frameworks.

Server Execution: At Line 2, the server randomly initializes the global model weights θ_0. At Line 3 it generates an augmented sample set $Z_{\text{aug}} = \{z^{(1)}, \ldots, z^{(n)}\}$ from the target record z. For each communication round $t = 1, \ldots, T$, the server first invokes ClientUpdate on each client (Lines 5–7) to collect the local models $\{\theta_k^t\}_{k=1}^K$. At Line 8 these are aggregated via weighted averaging to form the new global model θ_t. To perform the source inference attack, at Line 9 the server applies gradient ascent on θ_t with respect to both z and all samples in Z_{aug}, amplifying any differences in client-specific gradient contributions. Finally, at Lines 10–12 the server computes the average loss of each client's local model on z and Z_{aug}, and at Line 13 identifies the source client $i = \arg\min_k \bar{\ell}_k^T$. After T rounds, it returns the final global model θ_T and the inferred client index i.

ClientUpdate(θ): Upon receiving parameters θ at Line 16, each client k splits its local dataset D_k into mini-batches of size B (Line 17). Over E local epochs (Lines 18–22), it performs mini-batch SGD: for each batch b, it updates $\theta \leftarrow \theta - \eta \nabla_\theta \ell(b; \theta)$. After completing all epochs, at Line 23 the client returns the updated model parameters to the server.

Computational Complexity: Let D be the dimension of the model, $|D_k|$ the size of client k's dataset, and $n = |Z_{\text{aug}}|$.

- Local cost per client per round = $O(E \times \frac{|D_k|}{B} \times D)$,
- Server augmentation cost per round = $O((n+1) \times D)$,
- Server loss-evaluation cost per round = $O(K \times (n+1) \times D)$.

Over T rounds, the total time complexity is $O(T[K\,E\,\frac{|D_k|}{B} D + (n+1)D + K(n+1)D])$. Space complexity is dominated by storing the model ($O(D)$) and the augmented samples ($O(n \cdot \dim(z))$).

4 Experiment

In this section, we conduct a comprehensive evaluation of the ESIAs across various FL frameworks. Our experiments aim to assess the effectiveness, robustness, and generalizability of ESIAs in different settings. We compare our approach with baseline attacks in existing works, analyzing performance based on key evaluation metrics to demonstrate its advantages and practical implications.

4.1 Experiment Settings

Dataset	# of Records	# of Classes	Dimension of records
Synthetic	100k	10	60
MNIST	70k	10	1x28x28
CIFAR-10	60k	10	3x32x32
FEMNIST	80k	62	1x28x28
CIFAR-100	60k	100	3x32x32
Purchase	197.3k	100	600

Table 1: Summary of datasets used in experiments.

We follow the experimental setup with previous work [Hu *et al.*, 2021; Hu *et al.*, 2023], adopting standard configurations for datasets, models, and metrics to ensure consistency and comparability.

Datasets. The datasets used in our experiments are summarized in Table 1. To precisely control data heterogeneity, we create an independent and identically distributed (IID) synthetic dataset. Additionally, we use widely adopted datasets, including MNIST [McMahan *et al.*, 2017], CIFAR-10 [Smith *et al.*, 2017], FEMNIST [Caldas *et al.*, 2018], and CIFAR-100 [Bonawitz *et al.*, 2019], which serve as standard benchmarks to evaluate privacy leakage in federated learning.

Metrics. We evaluate ESIAs using the attack success rate (ASR), defined as the ratio of successful attacks to the total number of attacks, which quantifies the attack effectiveness.

Hyperparameter Settings. We configure the experiments as follows. For optimizer, we use Stochastic Gradient Descent with a learning rate of 0.01. We simulate 10 clients in federated learning, each with 100 target records for ESIAs. The communication rounds is set to 20, which is sufficient for ensuring global model convergence.

Factors Influencing ESIAs. Two key factors impact the effectiveness of ESIAs: the degree of data distribution heterogeneity and the number of local training epochs.

• *Data Distribution* α. In FL, client data is often non-IID. To simulate this, we use a Dirichlet distribution parameterized by α. Larger α values (e.g., 100) result in more homogeneous client data distributions, while smaller α values (e.g., 0.1) lead to highly skewed distributions.

• *Local Training Epochs* E. The number of local training epochs affects model retention. For FedSGD, we set $E = 1$, meaning only one local update per communication round. For

	Datasets	The ASR (%)											
		$\alpha = 100$				$\alpha = 1$				$\alpha = 0.1$			
		ESIAs	ASI	SIAs	RG	ESIAs	ASI	SIAs	RG	ESIAs	ASI	SIAs	RG
FedSGD	Synthetic	**21.3**	20.2	19.1	10.0	**37.8**	33.5	30.9	10.0	**67.4**	60.2	55.9	10.0
	Purchase	**18.2**	16.6	15.7	10.0	**38.4**	32.7	30.6	10.0	**73.5**	67.2	63.9	10.0
	MNIST	**14.5**	13.1	12.7	10.0	**30.8**	22.8	23.1	10.0	**68.5**	55.7	50.2	10.0
	CIFAR-10	**19.6**	16.8	17.6	10.0	**33.7**	31.4	28.5	10.0	**71.4**	63.8	58.3	10.0
FedAvg	Synthetic	**24.8**	22.4	18.9	10.0	**34.3**	30.1	28.5	10.0	**65.7**	54.6	51.7	10.0
	Purchase	**31.6**	30.1	28.2	10.0	**40.9**	35.3	34.8	10.0	**78.3**	69.4	66.2	10.0
	MNIST	**17.3**	15.2	13.5	10.0	**28.2**	23.2	22.1	10.0	**63.7**	53.8	42.3	10.0
	CIFAR-10	**56.7**	53.6	51.1	10.0	**62.8**	57.3	55.8	10.0	**74.3**	66.1	62.5	10.0

Table 2: The comparison study with ASI, SIAs and RG.

other FL frameworks, we use multiple local epochs, allowing models to capture local data patterns effectively.

Comparison Methods: To evaluate the effectiveness of the proposed ESIAs, we compare it with three existing baseline source inference methods: Randomly Guessing (RG), SIAs and Active Source Inference (ASI) [Zhang and Xia, 2024].

• *RG.* This straightforward baseline operates by randomly selecting a client as the source of a target record. Its expected ASR is $1/K$, where K represents the number of clients. Although simple and rudimentary, RG establishes a lower bound for attack performance. However, its effectiveness diminishes significantly as the number of clients increases, making it a minimal benchmark for comparison.

• *SIAs.* We adopt the approach proposed by Hu et al. [Hu *et al.*, 2023] as a foundational baseline. As a pioneering method in source inference attacks, SIAs have gained widespread adoption in related research. This technique provides a reliable and standard benchmark for evaluating attack performance, offering a consistent point of reference across studies.

• *ASI.* The Active Source Inference combines label flipping with supervised learning to exploit client-specific discrepancies in federated learning. By deliberately corrupting a subset of target data through label flips at the server, ASI triggers distinct test loss patterns in the global model for the source client. A supervised attack model, trained on these loss patterns, delivers high-precision source inference with minimal computational cost. While this approach introduces slight degradation in model performance, ASI consistently surpasses the other baselines, underscoring its superior effectiveness in source inference tasks.

4.2 Comparison Study

As illustrated in Table 2, the ASR of SIAs consistently surpasses the 10% random guessing baseline across all communication rounds and datasets, underscoring their effectiveness in deducing source information from training data records. In contrast, our proposed ESIAs exhibit superior performance, achieving elevated ASR across all evaluated scenarios. This improvement is driven by the integration of data augmentation and gradient ascent techniques, which enhance the distinguishability of source signals in target records across distinct clients, thereby strengthening the inference capabilities.

Moreover, ESIAs consistently outperforms the ASI method across all datasets. While ASI employs a label-flipping approach that amplifies source signals to a certain extent, our strategy, combining data augmentation with gradient ascent, demonstrates greater efficacy in achieving this objective. A key advantage of ESIAs lies in their adaptability, offering precise control over the trade-off between attack success and model utility. This is accomplished by adjusting the gradient ascent learning rate and the degree of data augmentation. For example, within the FedAvg framework with α = 0.1 on the Synthetic dataset, standard training yields a model accuracy of 91.65%. In comparison, our ESIA method achieves an accuracy of 88.51%, while ASI records 86.32%. By fine-tuning the gradient ascent learning rate and the scope of data augmentation, ESIA can elevate model accuracy to 90.87%, albeit with a reduction in ASR from 65.7% to 61.4%. Nevertheless, this adjusted ASR still exceeds the performance of ASI. Variations in ASR are observed across different FL frameworks and datasets, primarily due to disparities in the degree of local model overfitting to their respective training data. These differences and their wider implications will be elaborated upon in subsequent sections of this study.

4.3 Ablation Study

To systematically evaluate the influence of data distribution and the number of local training epochs on the effectiveness of ESIAs, we performed a comprehensive ablation study across several FL frameworks: FedSGD, FedAvg, FedMD, FedPer, and FedProx. Given that FedPer and FedProx are specifically designed to address non-IID data scenarios, we restricted our analysis of local training epochs to non-IID conditions. During the training process, we tracked and reported the highest ASR observed for each configuration. Due to its design, SIA in FedSGD (i.e., FedSGD-ESIA) is limited to a single local training epoch (E=1) per communication round, with results for higher epochs (e.g., E=5 or E=10) unavailable and marked as "–" in the tables.

Impact of Local Training Epochs. As illustrated in Table 3, an increase in the number of local training epochs (E) generally enhances the ASR for frameworks such as FedAvg-ESIA and FedMD-ESIA. For instance, in FedAvg-ESIA with $\alpha = 0.1$, the ASR on the Purchase dataset rises from 73.2% to 78.3% as E increases. This improvement can be attributed to the strengthened confidence of local models resulting from prolonged training on client-specific data. However, this trend is not consistent across all cases. On the MNIST dataset, for example, the ASR declines from 67.5% to 63.7% with extended training. This reduction stems from

	Datasets	The ASR (%) of ESIAs								
		$\alpha = 100$			$\alpha = 1$			$\alpha = 0.1$		
		$E = 1$	$E = 5$	$E = 10$	$E = 1$	$E = 5$	$E = 10$	$E = 1$	$E = 5$	$E = 10$
FedSGD-ESIA	Synthetic	21.3	–	–	37.8	–	–	67.4	–	–
	Purchase	18.2	–	–	38.4	–	–	73.5	–	–
	MNIST	14.5	–	–	30.8	–	–	68.5	–	–
	CIFAR-10	19.6	–	–	33.7	–	–	71.4	–	–
FedAvg-ESIA	Synthetic	22.4	23.3	24.8	33.5	32.1	34.3	66.8	64.4	65.7
	Purchase	19.8	24.9	31.6	37.4	41.2	40.9	73.2	71.2	78.3
	MNIST	16.6	16.2	17.3	26.7	27.1	28.2	67.5	65.2	63.7
	CIFAR-10	19.5	52.5	56.7	28.4	55.6	62.8	68.8	70.9	74.3
FedMD-ESIA	FEMNIST	16.4	18.5	19.1	24.6	26.2	26.8	52.7	55.1	58.6
	CIFAR-100	21.2	22.1	24.9	27.3	29.4	31.5	49.6	54.2	57.9

Table 3: The ablation study on data distribution and local epochs.

	Datasets	The ASR (%) of ESIAs		
		$\alpha = 0.1$		
		$E = 1$	$E = 5$	$E = 10$
FedPer-ESIA	Synthetic	65.3	69.2	73.8
	Purchase	68.9	72.6	77.3
	MNIST	61.6	65.1	71.7
	CIFAR-10	60.3	64.5	68.8
FedProx-ESIA	Synthetic	34.6	36.8	40.8
	Purchase	45.2	48.3	51.4
	MNIST	38.4	41.5	42.3
	CIFAR-10	42.0	44.3	47.3

Table 4: ESIAs in FedPer and FedProx.

	Dataset	The ASR (%)			
		$\alpha = 0.1, E = 10$			
		SIAs	**GA**	**DA**	**ESIAs**
FedAvg	Synthetic	51.7	55.6	61.4	65.7
	Purchase	66.2	69.8	72.6	78.3
	MNIST	42.3	44.7	60.1	63.7
	CIFAR-10	62.5	67.3	72.5	74.3

Table 5: The ablation study of GA and DA.

a dual effect: while prolonged training improves the fit to local data, it also enhances generalization to data from other clients, thereby reducing the disparities in prediction losses across clients and weakening the effectiveness of ESIAs.

Further analysis, presented in Table 4, reveals divergent behaviors between FedPer and FedProx under non-IID conditions. FedPer's personalized sub-models, which prioritize adaptation to local data, achieve higher ASR by capturing client-specific patterns. Conversely, FedProx imposes regularization constraints that penalize deviations of local models from the global model, promoting generalization and reducing overfitting. This results in smaller prediction loss differences across clients, slightly compromising local performance but significantly mitigating the vulnerability to ESIAs by limiting global model overfitting.

Impact of Non-IID Data Distribution. The degree of non-IID data distribution among clients markedly affects ESIA performance, as evidenced in Table 3. Across all frameworks and datasets, ASR increases as the non-IID characteristics become more pronounced. In highly non-IID settings, clients often hold training data predominantly from a single class, leading to local models that exhibit minimal prediction loss on their own data but substantial loss on data from other clients. These pronounced differences in prediction losses across clients enable the server to effectively execute ESIAs, accurately inferring the source of a given training record.

ESIA Components. By comparing Gradient Ascent (GA) and Data Augmentation (DA), the ablation study on core ESIA mechanisms indicates that DA is significantly more effective than GA in revealing client source information (see Table 5). The reason is that GA requires SIAs to identify clients that perform robustly on both the target record and its variants, thereby enhancing the stability of SIAs. This emphasis on robust identification is key to improving SIA's overall success rate.

In FL, factors like Non-IID data lead to local model overfitting. This occurs because models prioritize client-specific patterns, undermining global generalization. Such overfitting creates unique local behaviors that ESIAs exploit through data augmentation and gradient ascent to pinpoint the client tied to target records, posing a privacy threat.

5 Future Work

Despite its idealized and rare practical occurrence, enhancing ESIA ASR under IID assumptions remains worthy of in-depth study. We posit that relying solely on inferences from single prediction outcomes of a target record across various clients yields insufficient source information to accurately pinpoint its origin client. Therefore, considering other data types is necessary to expand source information dimensions. For instance, pre-classifying target data records using spatio-temporal features (e.g., time, location) can help aggregate records likely belonging to the same source client. This approach is promising in specific fields like healthcare, such as identifying clustered abnormal physiological indicators during a rare disease outbreak in a specific region, or tracking localized drug prevalence within a healthcare system.

6 Conclusion

In this paper, we propose enhanced source inference attacks using data augmentation and gradient ascent, significantly improving the ASR. ESIAs effectively exploit client-side target record disparities and adapt to varied data distributions in federated learning. Experimental results validate our approach's superiority over existing methods, showing substantial increases in attack success rates and highlighting the critical need for stronger privacy-preserving mechanisms in FL.

Acknowledgments

This work was supported in part by Jiangsu Provincial Major Project on Basic Research of Cutting-edge and Leading Technologies, under grant no. BK20232032.

References

[Arivazhagan *et al.*, 2019] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

[Bagdasaryan *et al.*, 2020] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, and Vitaly Shmatikov. How to backdoor federated learning. In *International Conference on Artificial Intelligence and Statistics*, pages 2938–2948. PMLR, 2020.

[Bonawitz *et al.*, 2019] Keith Bonawitz, Hubert Eichner, Wolfgang Grieskamp, Dzmitry Huba, Alex Ingerman, Vladimir Ivanov, Chloe Kiddon, Jakub Konečnỳ, Stefano Mazzocchi, Brendan McMahan, et al. Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems*, 1:374–388, 2019.

[Caldas *et al.*, 2018] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

[Devakumar *et al.*, 2020] Delan Devakumar, Geordan Shannon, Sunil S Bhopal, and Ibrahim Abubakar. Racism and discrimination in covid-19 responses. *The Lancet*, 395(10231):1194, 2020.

[Durmus *et al.*, 2021] Alp Emre Durmus, Zhao Yue, Matas Ramon, Mattina Matthew, Whatmough Paul, and Saligrama Venkatesh. Federated learning based on dynamic regularization. In *International conference on learning representations*, 2021.

[Feng *et al.*, 2024] Bingdao Feng, Di Jin, Xiaobao Wang, Fangyu Cheng, and Siqi Guo. Backdoor attacks on unsupervised graph representation learning. *Neural Networks*, 180:106668, 2024.

[Geiping *et al.*, 2020] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in neural information processing systems*, 33:16937–16947, 2020.

[Hu *et al.*, 2021] Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, and Xuyun Zhang. Source inference attacks in federated learning. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1102–1107. IEEE, 2021.

[Hu *et al.*, 2023] Hongsheng Hu, Xuyun Zhang, Zoran Salcic, Lichao Sun, Kim-Kwang Raymond Choo, and Gillian Dobbie. Source inference attacks: Beyond membership inference attacks in federated learning. *IEEE Transactions on Dependable and Secure Computing*, 21(4):3012–3029, 2023.

[Jin *et al.*, 2025] Di Jin, Yujun Zhang, Bingdao Feng, Xiaobao Wang, Dongxiao He, and Zhen Wang. Backdoor attack on propagation-based rumor detectors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 17680–17688, 2025.

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

[Khodak *et al.*, 2021] Mikhail Khodak, Renbo Tu, Tian Li, Liam Li, Maria-Florina F Balcan, Virginia Smith, and Ameet Talwalkar. Federated hyperparameter tuning: Challenges, baselines, and connections to weight-sharing. *Advances in Neural Information Processing Systems*, 34:19184–19197, 2021.

[Kumar *et al.*, 2023] Kummari Naveen Kumar, Chalavadi Krishna Mohan, and Linga Reddy Cenkeramaddi. The impact of adversarial attacks on federated learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):2672–2691, 2023.

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

[Li *et al.*, 2020a] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

[Li *et al.*, 2020b] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020.

[Li *et al.*, 2021a] Qinbin Li, Zeyi Wen, Zhaomin Wu, Sixu Hu, Naibo Wang, Yuan Li, Xu Liu, and Bingsheng He. A survey on federated learning systems: vision, hype and reality for data privacy and protection. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[Li *et al.*, 2021b] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pages 6357–6368. PMLR, 2021.

[Li *et al.*, 2025] Jiaxin Li, Marco Arazzi, Antonino Nocera, and Mauro Conti. Subject data auditing via source inference attack in cross-silo federated learning. *Journal of Information Security and Applications*, 90:104034, 2025.

[Lyu *et al.*, 2020] Lingjuan Lyu, Han Yu, and Qiang Yang. Threats to federated learning: A survey. *arXiv preprint arXiv:2003.02133*, 2020.

[Lyu *et al.*, 2022] Lingjuan Lyu, Han Yu, Xingjun Ma, Chen Chen, Lichao Sun, Jun Zhao, Qiang Yang, and S Yu Philip. Privacy and robustness in federated learning: Attacks and defenses. *IEEE transactions on neural networks and learning systems*, 2022.

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

[Melis *et al.*, 2019] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (S&P)*, pages 691–706. IEEE, 2019.

[Nasr *et al.*, 2019] Milad Nasr, Reza Shokri, and Amir Houmansadr. 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 (S&P)*, pages 739–753. IEEE, 2019.

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

[Smith *et al.*, 2017] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.

[Tolpegin *et al.*, 2020] Vale Tolpegin, Stacey Truex, Mehmet Emre Gursoy, and Ling Liu. Data poisoning attacks against federated learning systems. In *Computer security–ESORICs 2020: 25th European symposium on research in computer security, ESORICs 2020, guildford, UK, September 14–18, 2020, proceedings, part i 25*, pages 480–501. Springer, 2020.

[Wang *et al.*, 2020] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020.

[Wang *et al.*, 2022] Zhibo Wang, Yuting Huang, Mengkai Song, Libing Wu, Feng Xue, and Kui Ren. Poisoning-assisted property inference attack against federated learning. *IEEE Transactions on Dependable and Secure Computing*, 2022.

[Wei *et al.*, 2025] Jiaheng Wei, Yanjun Zhang, Leo Yu Zhang, Chao Chen, Shirui Pan, Kok-Leong Ong, Jun Zhang, and Yang Xiang. Extracting private training data in federated learning from clients. *IEEE Transactions on Information Forensics and Security*, 2025.

[Xie *et al.*, 2022] Yuanyuan Xie, Bing Chen, Jiale Zhang, and Wenjuan Li. Algans: Enhancing membership inference attacks in federated learning with gans and active learning. In *2022 IEEE International Symposium on Product Compliance Engineering-Asia (ISPCE-ASIA)*, pages 1–6. IEEE, 2022.

[Xu *et al.*, 2020] Yongchao Xu, Liya Ma, Fan Yang, Yanyan Chen, Ke Ma, Jiehua Yang, Xian Yang, Yaobing Chen, Chang Shu, Ziwei Fan, et al. A collaborative online ai engine for ct-based covid-19 diagnosis. *medRxiv*, 2020.

[Zhang and Xia, 2024] Lening Zhang and Hui Xia. Active source inference attack based on label-flipping in federated learning. In *2024 IEEE 23rd International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1675–1680. IEEE, 2024.

[Zhao *et al.*, 2020] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*, 2020.

[Zhou *et al.*, 2022] Chunyi Zhou, Yansong Gao, Anmin Fu, Kai Chen, Zhiyang Dai, Zhi Zhang, Minhui Xue, and Yuqing Zhang. Ppa: Preference profiling attack against federated learning. In *Network and Distributed Systems Security Symposium 2022*. Internet Society, 2022.

[Zhu *et al.*, 2019] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in Neural Information Processing Systems*, 32, 2019.