Privacy Leaks by Adversaries: Adversarial Iterations for Membership Inference Attack

Jing Xue[1], Zhishen Sun[1], Haishan Ye[1,3*], Luo Luo[2], Xiangyu Chang[1], Guang Dai[3]

[1]Xi'an Jiaotong University
[2]Fudan University
[3]SGIT AI Lab, State Grid Corporation of China

Abstract

Membership inference attack (MIA) has become one of the most widely used and effective methods for evaluating the privacy risks of machine learning models. This attack aims to determine whether a specific sample is part of the model's training set by analyzing the model's output. While traditional membership inference attacks focus on leveraging the model's posterior output, such as confidence on the target sample, we propose `IMIA`, a novel attack strategy that utilizes the process of generating adversarial samples to infer membership. We propose to infer the member properties of the target sample using the number of iterations required to generate its adversarial sample. We conduct experiments across multiple models and datasets, and our results demonstrate that the number of iterations for generating an adversarial sample is a reliable feature for membership inference, achieving strong performance both in black-box and white-box attack scenarios. This work provides a new perspective for evaluating model privacy and highlights the potential of adversarial example-based features for privacy leakage assessment.

Code — https://github.com/Xuejing0203/Imia2

Introduction

Machine learning has widespread applications in many fields, such as autonomous driving (Aoki et al. 2023; Yu et al. 2024), medical (Dixit 2021) and financial systems (Chatzis et al. 2018; Samitas, Kampouris, and Kenourgios 2020). Training a model requires collecting a large amount of data and aims to help the model learn knowledge that generalizes well from the training data. For example, a hospital may train a diagnostic model using patients' CT scans and treatment outcomes. While, this model is used to assist in diagnosis and treatment, several studies (Carlini et al. 2022b; Salem et al. 2018; Shokri et al. 2017) have shown that neural network models tend to remember their training data and an adversary can exploit this weakness to launch membership inference attack (MIA) (Carlini et al. 2022a; Choquette-Choo et al. 2021; Song and Mittal 2021). In such case, an adversary could infer whether a particular patient's record was used during the training process - potentially disclosing sensitive information like diagnosis results.

*Corresponding author.

As a fundamental method to evaluate the privacy risk of machine learning models, membership inference attack (MIA) has received a lot of attention in recent years (Bertran et al. 2024; Debenedetti et al. 2024; Shokri et al. 2017; Watson et al. 2022). Specifically, given a target model, an adversary aims to know if a target sample was part of the model's training set (being a member) or not (being a non-member). Studying attack methods such as MIA is important for understanding and evaluating the privacy risk of machine learning models.

Membership inference attacks (MIA) typically fall into two categories. Distribution-based MIA methods exploit distributional differences between the training and test data, but they typically require large datasets and shadow models, making them resource-intensive (Carlini et al. 2022a; Chaudhari et al. 2023; Tramèr et al. 2022). In contrast, metric-based MIA methods infer membership from the model's output, such as confidence scores, without access to the training data or shadow models (Chen et al. 2022; Song, Shokri, and Mittal 2019; Yeom et al. 2018). However, these methods are limited to scenarios where the model exposes soft outputs (e.g., probabilities). For instance, the Softmax Response attack is effective only when the target model outputs confidence values and fails when only hard labels are available.

These limitations prompt us to ask: whether there exists a universal method that can solve these limitations, and remain effective in black-box as well as white-box scenarios without requiring extensive data or computation.

In this paper, we affirmatively answer this question by proposing a novel membership inference attack method, Iterations for Membership Inference Attack (`IMIA`), from the lens of adversarial samples' generation. Our key observation is that member samples, being further to the decision boundary, generally require more iterations to generate adversarial examples than non-member samples. `IMIA` leverages this iteration gap across different settings—including white-box and black-box—by employing suitable adversarial attack strategies such as HopSkipJumpAttack, SimBA, and PGD (Chen, Jordan, and Wainwright 2020; Guo et al. 2019; Madry et al. 2018). Unlike prior work that relies on shadow models or access to similar data distributions (Carlini et al. 2022a; Tramèr et al. 2022), IMIA operates without requiring the training set, making it lightweight and broadly

applicable.

In general, our contributions are summarized as follows:

1. In this study, we propose a novel member inference attack method IMIA to infer whether the data belongs to the training set by analyzing the number of iterations required to generate adversarial samples. Different from traditional attack methods based on the posterior output of the target model, IMIA focuses on the generation process of adversarial samples, providing a tool to evaluate privacy leaks from the perspective of the internal operation of the model.
2. IMIA does *not* require any training data and training shadow models. The target sample is sufficient for IMIA to execute the attack. This strategy leverages the number of iterations required to generate adversarial samples from the target sample for MIA instead of posterior outputs of the target model.
3. IMIA is highly adaptable and universal. Our proposed method IMIA can be exploited in all settings compared with previous methods that can only be used in one specific situation. We have conducted experiments on multiple models and datasets, covering different network architectures and data distributions. Experimental results show that our method can effectively evaluate the privacy leakage risk of the model under both black-box and white-box settings.

Background and Related Work

In this section, we review research on membership inference attack, adversarial samples, and existing methods that we use as baselines.

Membership Inference Attack

Membership inference attack has achieved great attention because it revealed that machine learning models have serious risks of privacy leakage and remember its training data (Bertran et al. 2024; Carlini et al. 2019; Nasr et al. 2025; Prashanth et al. 2025; Tobaben et al. 2024). In membership inference attack, given the target sample x, the adversary aims to infer whether this sample is in the training set $\mathrm{D_{tr}}$ of the target model f_θ. As a result, membership inference attack can be seen as a binary classification privacy game. The participants in this game are the challenger $\mathcal{C}$ and the adversary $\mathcal{A}$. The game process can be broken down into several steps:

1. The challenger samples a training set $\mathrm{D_{tr}} \sim \mathbb{D}$, and trains the target model f_θ.
2. The challenger randomly chooses a bit $b \in \{0, 1\}$. If $b = 0$, he will choose the target sample $(x, y) \in \mathbb{D}$ where y is the ground-truth label of the target sample x, but (x, y) is not in $\mathrm{D_{tr}}$. If $b = 1$, he will choose the target sample in $\mathrm{D_{tr}}$ directly.
3. The challenger sends the target sample to the adversary and allows the adversary to query the target model.
4. The adversary gets the target sample and returns a bit $\hat{b}$ by querying the target model. If $\hat{b} = b$, the adversary will win this game.

Based on the adversary's ability to access the target model, MIA can be divided into two categories:

Black-box Membership Inference. In the black-box setting, the adversary can only access the posterior output of the model (Nasr, Shokri, and Houmansadr 2019). This is a true scenario in the real world. In the black-box case, the attack methods can also be divided into two categories: the first is a distribution-based MIA (Carlini et al. 2022a; Chaudhari et al. 2023; Debenedetti et al. 2024; Tramèr et al. 2022), in which the adversary needs to train additional shadow models as a proxy to mimic the target model. These shadow models require amount of data in the same distribution as the target sample and use the same model framework.

The second is the metric-based MIA, in which the adversary does not need to train additional shadow models. The adversary only uses the posterior output of the target model and designs a metric $\mathrm{Me}(\cdot)$, such as Softmax Response (Nasr, Shokri, and Houmansadr 2019; Song, Shokri, and Mittal 2019), Prediction Entropy (Salem et al. 2018; Yeom et al. 2018) and Modified Entropy (Song and Mittal 2021). Specifically, Softmax Response (Song, Shokri, and Mittal 2019) computes the output probabilities of the target model $f_\theta(x)$ for the input sample x, obtaining the predicted probability $f_\theta(x)_i$ for each class i, and compares it with a preset threshold τ. If the maximum predicted probability exceeds the threshold, the adversary infers that the sample belongs to the training set. Formally,

$$I_{soft}(f_\theta, (x, y)) = \mathbb{1}\{\max f_\theta(x)_i \geq \tau\}$$

Salem et al. (2018) proposed to use the Prediction Entropy to conduct MIA. Prediction Entropy measures the uncertainty of the model's prediction and they thought that the prediction entropy of the member data would be close to 0 and a larger entropy value to the non-member data on the target model. This is formally expressed as:

$$I_{ent}(f_\theta, (x, y)) = \mathbb{1}\{-\sum_i f_\theta(x)_i \log(f_\theta(x)_i) \leq \tau\}$$

Besides, Song and Mittal (2021) proposed Modified Entropy that considered the ground-truth label of the target sample by decreasing the uncertainty on the true label of the target sample and increasing the uncertainty on the wrong label. It can be computed through:

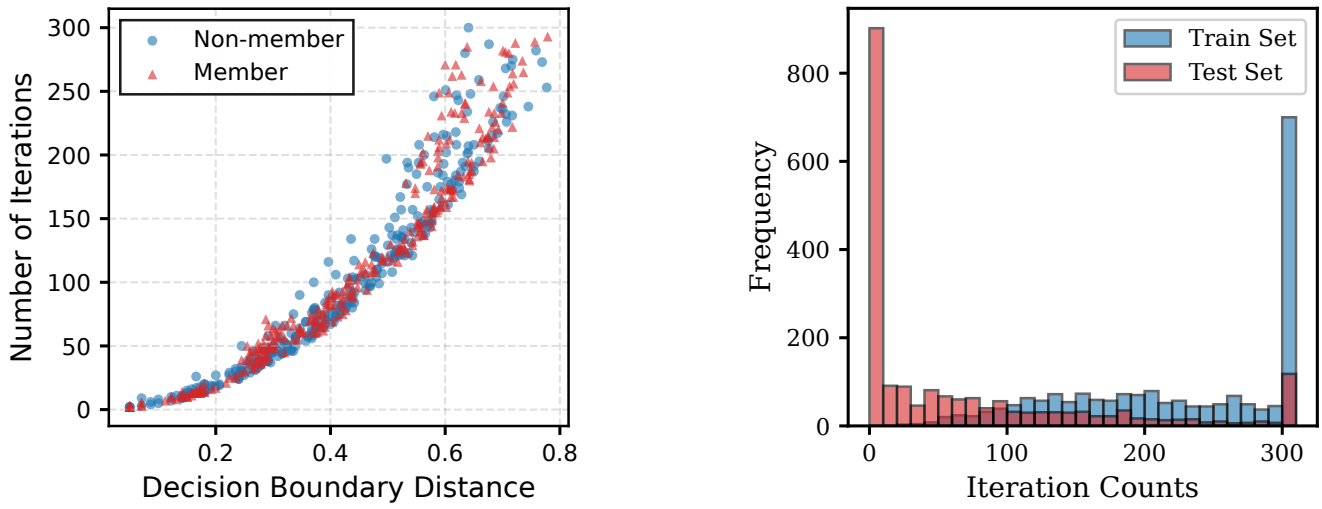
$$\begin{aligned} Mentr(f_\theta(x), y) = &-(1 - f_\theta(x)_y)\log(f_\theta(x)_y) \\ &-\sum_{i \neq y} f_\theta(x)_i \log(1 - f_\theta(x)_i) \end{aligned}$$

They inferred a target sample in $\mathrm{D_{tr}}$ according to:

$$I_{Mentr}(f_\theta, (x, y)) = \mathbb{1}\{Mentr f_\theta(x, y) \leq \tau\}$$

That is if the modified entropy value of the target sample is lower than the preset threshold, it will be recognized as a member.

White-box Membership Inference. In the white-box setting, the adversary can not only get the posterior output of the target model but also the internal parameters of the target model, like the loss and gradient information during the progress of the target model training (Carlini and Wagner



(a) a

(b) b

Figure 1: (a) Scatter diagram showing the relationship between the distance from samples to the decision boundary and the number of iterations required to generate adversarial examples using SimBA for ResNet trained on CIFAR10. (b) Histogram about the number of iterations per-sample over 2k samples from the training set (blue) and the same number from the test set. The adversarial samples are generated using SimBA for ResNet trained on CIFAR100.

2017; Goodfellow, Shlens, and Szegedy 2014). Yeom et al. (2018) used the internal weights of the model and the loss of the target sample on the model to conduct MIA. If an adversary can get access to the logits output of the target model, he can conduct MIA through:

$$I_{loss}(f_\theta, (x, y)) = \mathbb{1}\{-l(f_\theta(x), y) \leq \tau\}$$

where the loss function will be cross-entropy loss. The sample x which has lower loss value in the training set D_{tr}. This capability allows an adversary to obtain more sensitive information of the victim model, but this capability is rarely available in the real world.

Adversarial Sample

In a deep neural network, adversarial samples are inputs intentionally perturbed with small but deliberate perturbations that can cause deep neural networks to make incorrect predictions. Goodfellow, Shlens, and Szegedy (2014) first revealed the vulnerability of neural networks to such inputs. Then, numerous attack methods have been proposed, including white-box approaches like PGD (Madry et al. 2018) and CW (Carlini and Wagner 2017), as well as black-box approaches like SimBA (Guo et al. 2019), which provide effective strategies for generating adversarial examples.

In the background of MIA, the characteristics of adversarial samples are used to infer the privacy of the model's training data. Song, Shokri, and Mittal (2019) used the confidence output on adversarial samples to judge the member attributes of the target samples. They believed that due to the robustness of the model, the adversarial samples generated from the member data show relatively stable prediction results on the model. Afterwards, Choquette-Choo et al. (2021) found that the distance from the adversarial sample to the model's decision boundary can be directly used to carry out MIA and had a great performance under the black-box setting. They judged a sample as a member if the distance from the adversarial sample to its decision boundary is larger than a preset threshold:

$$I_{dis}(f_\theta, (x, y)) = \mathbb{1}\{d(x, \hat{x}) \geq \tau\}$$

Del Grosso et al. (2022) also analyzed MIA from this perspective.

In our paper, we carry out the metric-based black-box membership inference from the lens of the generation of adversarial samples. Different from prior works(Choquette-Choo et al. 2021), we do not measure the boundary distance between the adversarial sample and its decision boundary but record the number of iterations during the process of generating adversarial samples. Softmax Response, Prediction Entropy, and Modified Entropy as typical representatives in the score-based metric attacks, we choose these as our baselines. In the case where the target model only outputs hard labels, we choose the boundary distance attack as our baseline because boundary distance has a great performance in the metric-based condition.

Membership Inference Attack During Adversarial Examples Iteration

In this section, we discuss the methodology of this paper, which aims to reveal privacy risks of the target model from the lens of adversarial samples' generation process.

Motivation

While prior MIA methods often rely on confidence scores or boundary distance (Choquette-Choo et al. 2021; Song, Shokri, and Mittal 2019), we find that boundary distance may not reliably distinguish members from non-members because different samples can have similar distance regardless of membership, as shown in Figure 1(a). We plotted a scatter diagram showing the relationship between the distance from samples to the decision boundary and the number of iterations required to generate adversarial samples.

Instead, we observe that the member samples generally require more iterations to generate adversarial samples than non-members. We use "SimBA" (Guo et al. 2019) to generate adversarial samples and record the number of iterations to generate its adversarial sample for each sample. The histogram is shown in Figure 1(b): the samples from the training set (blue, member) need more iterations than those from the test set (pink, non-member).

This insight reveals a new, consistent signal for membership inference and motivates our method, IMIA, which leverages the number of iterations required to generate an adversarial sample for a target sample. If this number exceeds a preset threshold, this target sample will be inferred as a member; otherwise, as a non-member. The overall framework of IMIA is shown in Figure 2.

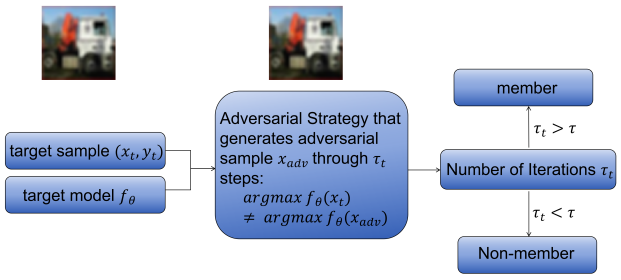


Figure 2: Diagrammatic sketch for IMIA to conduct MIA.

Methodology

Given the target model and the target images, the adversary can choose an adversarial strategy $\mathcal{S}$ in SimBA (Guo et al. 2019), HopSkipJumpAttack (Chen, Jordan, and Wainwright 2020) and PGD (Madry et al. 2018) based on different MIA settings to generate adversarial samples and measure the number of iterations during this process.

Score-based black-box attacks. Adversary can obtain the full probability output of the target sample. As a result, we choose SimBA (Guo et al. 2019) to conduct adversarial attack which provides a simple but efficient strategy to change the target sample's output. The optimization goal in SimBA is to minimize the probability on the true label y of the target sample x :

$$\min_{\delta} \quad p_{f_\theta}(y|x+\delta)$$
$$\text{subject to:} \quad \|\delta\|_2 < d,\ \text{queries} \leq M,$$

where δ represents the perturbations and M is the budget for the number of queries to the target model. SimBA solves this problem through randomly selecting a predefined orthonormal basis and either adding or subtracting it from the target sample according to the confidence scores which are checked if the target sample moves toward the decision boundary. During the process of generating adversarial samples for the target sample, we are concerned about the number of iterations after getting adversarial samples successfully.

Decision-based black-box attacks. Different from score-based black-box attacks, the adversary can only obtain the label of the target input without any other information. In this case, we choose "HopSkipJumpAttack" (Chen, Jordan, and Wainwright 2020) whose performance is close to the white-box attack. Given the target image (x, y), adversary starts from randomly choosing a point x' that is not classified to label y by the target model and walks along the decision boundary to minimize the distance between the original image x to its adversarial image x'. In our methodology, we measure the number of iterations that adversarial samples can satisfy our request.

White-box attacks. In the white-box setting, We choose Projected Gradient Descent (PGD) (Madry et al. 2018) to generate adversarial samples. Formally, adversarial sample x_{adv} is generated by:

$$x_{adv} = \text{Clip}_{x,\epsilon}(x_N^{adv} + \alpha\, \text{sign}\left(\nabla_x J(x_{N+1}^{adv}, y)\right)$$

In IMIA, we will choose one of them to generate adversarial samples according to different MIA settings. The pseudocode for IMIA is listed in Algorithm 1. As shown in Algorithm 1, given a target model f_θ, a target sample (x_t, y_t), an adversarial attack strategy S, and a threshold τ, IMIA first computes the number of iterations τ_t required by S to generate a successful adversarial example. The adversarial strategy S is chosen based on the attack setting including PGD, SimBA and HopSkipJumpAttack as we described before. If $\tau_t \geq \tau$, the sample is classified as a *member*; otherwise, it is classified as a *non-member*.

Algorithm 1: IMIA

Require: Target model f_θ, target sample (x_t, y_t), adversarial strategy $\mathcal{S}$, threshold τ

$\tau_t \leftarrow \mathcal{S}(f_\theta, (x_t, y_t))$

Ensure: Number of iterations τ_t for generating adversarial sample

if $\tau_t \geq \tau$ **then**
  return $\mathbb{1}(\tau_t \geq \tau)$ ▷ Classify as member
else
  return 0 ▷ Classify as non-member
end if

Strategy	ResNet		ResNeXt		VGG		DenseNet	
	AUROC ↑	Accuracy ↑	AUROC	Accuracy	AUROC	Accuracy	AUROC	Accuracy
CIFAR100								
Softmax	88.91± 0.13	84.57± 0.15	63.66± 0.09	60.42± 0.10	63.73± 0.12	59.94± 0.18	67.70± 0.11	63.26± 0.14
Entropy	88.97± 0.08	84.59± 0.11	64.22± 0.13	54.42± 0.07	64.95± 0.14	61.26± 0.12	68.78± 0.16	64.00± 0.11
Mentr.	**89.24± 0.15**	**85.43± 0.16**	68.42± 0.10	64.48± 0.12	**69.61± 0.14**	**65.41± 0.13**	73.19± 0.12	67.90± 0.18
IMIA(ours)	89.11± 0.29	83.62± 0.29	**68.93± 0.58**	**64.82± 0.55**	67.36± 0.64	64.68± 0.55	**73.80± 0.48**	**68.91± 0.40**
CIFAR10								
Softmax	70.97± 0.05	65.84± 0.15	74.02± 0.13	68.05± 0.19	64.15± 0.16	60.60± 0.15	69.21± 0.15	64.17± 0.14
Entropy	71.05± 0.13	65.87± 0.29	**74.35± 0.14**	**68.24± 0.17**	64.39± 0.15	60.70± 0.12	69.51± 0.16	64.27± 0.14
Mentr.	71.95± 0.12	66.17± 0.16	73.54± 0.15	67.95± 0.11	64.56± 0.12	60.58± 0.14	69.44± 0.15	64.23± 0.12
IMIA(ours)	**74.43± 0.15**	**68.35± 0.13**	73.34± 0.09	67.77± 0.15	**66.53± 0.13**	**62.87± 0.15**	**75.20± 0.09**	**68.97± 0.15**
STL10								
Softmax	56.49± 0.13	55.53± 0.15	61.68± 0.17	59.43± 0.11	57.06± 0.16	56.63± 0.12	72.97± 0.12	68.52± 0.13
Entropy	55.91± 0.18	55.28± 0.19	62.08± 0.12	59.72± 0.11	57.75± 0.14	57.61± 0.13	73.22± 0.13	68.59± 0.15
Mentr.	61.05± 0.11	59.02± 0.13	**70.84± 0.18**	**66.37± 0.10**	**62.68± 0.14**	**59.92± 0.12**	78.41± 0.11	73.97± 0.09
IMIA(ours)	**61.26± 0.15**	**59.70± 0.12**	69.25± 0.15	66.35± 0.14	61.32± 0.08	59.87± 0.14	**81.18± 0.12**	**75.41± 0.11**

Table 1: Membership inference results for different score-based attacks on CIFAR10, CIFAR100, and STL10 datasets. We choose "SimBA" to generate adversarial samples for measuring the number of iterations for target samples. The evaluation set is composed of 3k samples from the training set and 3k samples from the test set of the target model, and we repeat this procedure 20 times. The inference accuracy(%) and AUROC(%) are reported in the average value with standard deviation.

In this paper, we consider three different attack strategies for different MIA settings. Note that we do not propose a novel method for adversarial attacks, instead, we only care about how the adversarial sample is generated from the original target sample.

Evaluation

In this section, we will evaluate the effectiveness and universality of our algorithm.

Experiment Setup

Datasets. In our experiment, we consider three different datasets which are all common in image recognition tasks. **CIFAR10** and **CIFAR100** all include 60k images and can be split into 10 and 100 classes respectively. In PyTorch, CIFAR10 and CIFAR100 are divided into 50k images in the training dataset and others are in the test dataset. **STL-10** includes 5k images in the training set and 8k images in its test set. In our experiment, we use all samples from the training dataset to train the target model for each model architecture, and samples in the test set which do not participate in the training process are used to validate the model's accuracy.

Target Model. In our experiments, we use four different model architectures: ResNet50, VGG19, ResNeXt29_2x64d, and DenseNet121. These models represent a diverse set of machine learning architectures. We train each model for 100 epochs in every dataset during the training process to ensure that the models are sufficiently trained and can produce meaningful outputs for membership inference attack.

Metrics. We use a balanced evaluation set to evaluate our method and report the inference accuracy, AUROC scores, and the false positive rate (fpr) under different true positive rate (tpr). The inference accuracy considers both the true positive rate and the false positive rate and gives 50% if the adversary guesses randomly. Area Under the Receiver Operating Characteristic Curve (AUROC) is the area under ROC curve, which is obtained by plotting the ratio of TPR to FPR at different thresholds. The closer the AUROC value is to 1, the better the attack performance.

In a balanced evaluation set, our evaluation set consists of 3k samples from the target model's training set as member samples and uniformly selects the same number of samples from the test set as non-member samples. We repeat this many times to get different combinations of evaluation sets and finally report results on average.

In our paper, IMIA is a metric-based universal attack method which does not need any training data or shadow models, so to ensure a fair comparison, we also choose the metric-based method as our baselines instead of these methods using shadow models.

Results

We analyze our results in the black-box and the white-box settings separately. In the black-box setting, there are two types: one is score-based attack where the target model outputs the confidence and labels at the same time; the other is the target model that only outputs hard labels. In the white-box setting, adversaries have access to the loss during the process of generating adversarial samples. Specifically, `IMIA` resorts to generate adversarial samples like "PGD" (Madry et al. 2018; Tramèr et al. 2017) for white-box MIA, "SimBA" (Guo et al. 2019) for black-box MIA and "HopSkipJumpAttack" for hard labels MIA.

Score-based Setting In the score-based setting, the target model outputs the confidence and labels at the same time. In this case, we choose "SimBA" (Guo et al. 2019) as our strategy to generate corresponding adversarial samples of

Strategy	ResNet		ResNeXt		VGG		DenseNet	
	AUROC ↑	Accuracy ↑	AUROC	Accuracy	AUROC	Accuracy	AUROC	Accuracy
CIFAR100								
Boundary	**86.12±0.35**	**81.59±0.43**	**69.49±0.29**	**66.90±0.34**	63.29±0.33	61.90±0.26	69.72±0.32	66.90±0.33
IMIA(ours)	85.95±0.27	81.39±0.25	69.20±0.21	66.70±0.25	**63.39±0.31**	**62.39±0.30**	**70.03±0.22**	**67.20±0.22**
CIFAR10								
Boundary	72.84±0.30	66.70±0.25	**70.43±0.26**	**65.60±0.31**	66.50±0.23	63.0±0.21	69.72±0.15	66.00±0.22
IMIA(ours)	**73.12±0.25**	**67.39±0.27**	69.77±0.25	65.10±0.23	**69.31±0.21**	**65.10±0.21**	**71.81±0.19**	**66.90±0.17**
STL10								
Boundary	61.45±0.27	60.24±0.31	70.82±0.35	67.34±0.31	63.78±0.24	61.60±0.20	82.42±0.13	75.29±0.17
IMIA(ours)	**62.00±0.19**	**60.41±0.22**	**72.21±0.25**	**69.38±0.20**	**63.93±0.18**	**61.63±0.24**	**83.36±0.19**	**76.57±0.21**

Table 2: Membership inference results for different decision-based attacks that only output hard labels on CIFAR10, CIFAR100 and STL10 datasets.

Strategy	ResNet		ResNeXt		VGG		DenseNet	
	AUROC ↑	Accuracy ↑	AUROC	Accuracy	AUROC	Accuracy	AUROC	Accuracy
CIFAR100								
Loss	89.24± 0.24	85.68± 0.26	68.47± 0.46	65.14± 0.51	69.95± 0.59	65.87± 0.60	73.31± 0.43	68.42± 0.46
Boundary	88.37± 0.31	82.82± 0.26	66.16±0.45	63.97±0.49	64.69±0.50	62.62±0.51	66.43 ±0.40	64.46±0.43
IMIA(ours)	**96.12± 0.21**	**90.54± 0.28**	**69.31± 0.54**	**65.82± 0.51**	**71.47± 0.45**	**68.55± 0.42**	**73.41± 0.40**	**68.91± 0.39**
CIFAR10								
Loss	72.26± 0.49	66.61± 0.45	74.33± 0.35	68.96± 0.39	65.09± 0.36	61.72± 0.44	69.94± 0.35	64.79± 0.32
Boundary	**75.45±0.47**	**69.03±0.45**	74.19±0.37	68.62±0.37	65.94±0.40	61.67±0.42	75.69±0.33	69.82±0.29
IMIA(ours)	74.49± 0.46	68.97± 0.39	**74.45± 0.34**	**69.83± 0.31**	**66.96± 0.35**	**62.67± 0.32**	**76.29± 0.33**	**70.15± 0.31**
STL10								
Loss	61.52± 0.45	59.20± 0.33	**71.01± 0.45**	**66.39± 0.42**	**62.94± 0.49**	**59.68± 0.36**	78.57± 0.46	74.06± 0.45
Boundary	60.29±0.43	58.75±0.32	68.38±0.50	65.25±0.47	59.66±0.45	59.05±0.35	81.78±0.46	75.63±0.47
IMIA(ours)	**61.94± 0.47**	**59.64± 0.36**	70.07± 0.53	66.19± 0.51	61.46± 0.52	59.61± 0.49	**82.19± 0.61**	**75.65± 0.50**

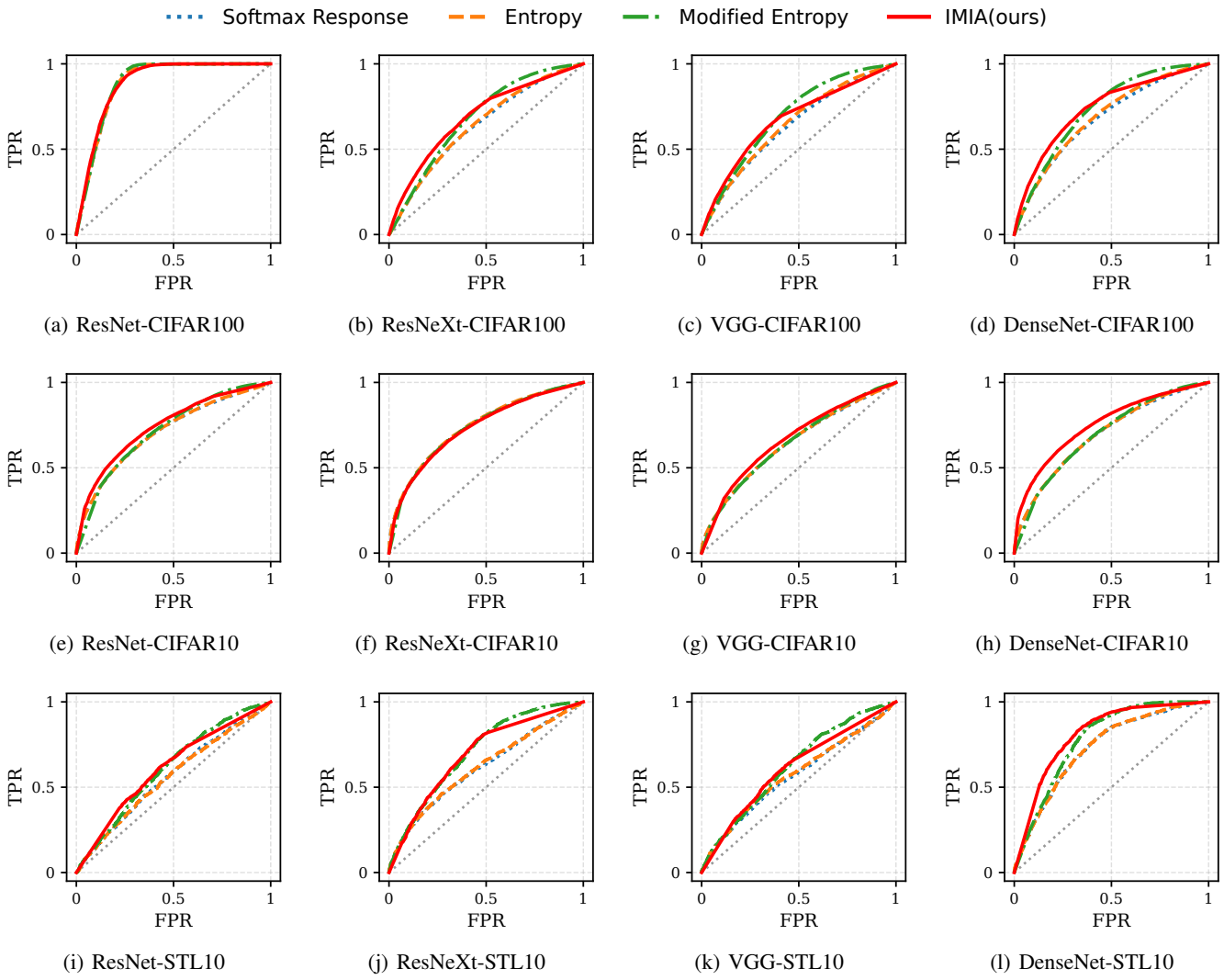
Table 3: Membership inference results for Loss and our method on CIFAR10, CIFAR100, and STL10 in the white-box setting.

the original samples. Softmax Response (Song, Shokri, and Mittal 2019), Prediction Entropy (Salem et al. 2018) and Modified Entropy (Song and Mittal 2021) as typical representatives in the score-based metric membership inference attack, we choose these as our baselines.

Table 1 shows the results of our attacks and comparisons between IMIA and other baselines in the score-based setting. The results show our strategy that depending on the number of iterations performs well in distinguishing member samples and non-member samples. For example, for our proposed attack against CIFAR10 DenseNet classifier, the membership inference AUROC is increased from 69.21 to 75.20 on average and the accuracy is increased from 64.17 to 68.97. In other words, our strategy can effectively reveal the privacy risk of the target model during the process of generating adversarial samples. Figure 3 show the false positive rate under different true positive rate corresponding to Table 1. The figures illustrate how the false positive rate varies as the true positive rate changes. Our proposed attack has a relatively lower fpr than others.

Decision-based Setting In another black-box setting called decision-based attack, the target model only outputs hard labels. In this case, we use "HopSkipJumpAttack" (Chen, Jordan, and Wainwright 2020) strategy to generate adversarial samples. We then compare our attack with the " Boundary " method (Choquette-Choo et al. 2021) which has strong performance when the target model only outputs hard labels. The "Boundary" measures the distance from the adversarial samples to their decision boundary. Our IMIA measures the number of iterations during the generation of adversarial samples. The comparison results are shown in Table 2. We can observe that IMIA has advantages over "Boundary" method. For example, the membership inference AUROC and accuracy in STL10 are all higher than "Boundary" method for all classifiers. Furthermore, for the VGG model on the CIFAR10, our IMIA achieves an accuracy about 2.8% higher than "Boundary". Though IMIA is simple, in this most difficult situation, it can still work efficiently.

White-box Setting We also evaluate IMIA in the white-box setting and show the comparison in Table 3 where the best results in each case are in bold. In the white-box setting, "Loss" method (Yeom et al. 2018) is one of the most common methods to measure the degree of privacy leaks,



(a) ResNet-CIFAR100 (b) ResNeXt-CIFAR100 (c) VGG-CIFAR100 (d) DenseNet-CIFAR100

(e) ResNet-CIFAR10 (f) ResNeXt-CIFAR10 (g) VGG-CIFAR10 (h) DenseNet-CIFAR10

(i) ResNet-STL10 (j) ResNeXt-STL10 (k) VGG-STL10 (l) DenseNet-STL10

Figure 3: ROC curve on MIA for the combination of different models on CIFAR100, CIFAR10 and STL-10. They are drawn on the balanced evaluation set and correspond to Table 1.

so we also use "Loss" method as baseline and compute the cross entropy loss to quantify the privacy risks associated with the target model. At the same time, we also compare the results of "Boundary" method. In this setting, we use "PGD" (Madry et al. 2018) methodology to generate adversarial samples. Table 3 shows that IMIA surpasses the "Loss" method both in the inference accuracy and AUROC in most classifiers and datasets. Especially, when applied to the DenseNet architecture on CIFAR10 dataset, IMIA can achieve an accuracy over 5% higher than the "Loss" method.

Summary

In three different application conditions, the increasing inference accuracy and AUROC value prove that our methodology IMIA has great performance in distinguishing the member data and the non-member data. Even in the most difficult situation, our methodology can still work well. All results show that IMIA has great adaptive ability and can be applied in both white-box and black-box settings without knowing data from the training set.

Conclusion

In this paper, We propose a universal membership inference attack method, called IMIA, which performs the membership inference attack from the perspective of adversarial samples' generation process. The key idea of IMIA is to measure the number of iterations by the process of generating the adversarial samples, and use this metric to infer whether the target samples belong to the model's training set or not. IMIA with different adversarial strategies can be applied in different settings. Accordingly, we conduct experiments in different MIA settings and on different datasets such as CIFAR10, CIFAR100 and STL10 datasets under dif-

ferent model architectures. Our experiment results show that our proposed methodology has great performance in different situations with higher AUROC values and inference accuracy compared to the other metric-based MIA algorithms. All experiments highlight the superior performance of IMIA and prove that IMIA is universal and adaptable in most settings to detect the privacy risk of the target model while requiring fewer computational resources, making it a more efficient choice for MIA.

Acknowledgments

This work was conducted at the SGIT AI LAB. We thank the members of the lab for their support and helpful discussions. Haishan Ye's work was supported by the National Key Research and Development Project of China under Grant 2022YFA1004002 and National Natural Science Foundation of China under Grant 72471185.

References

Aoki, S.; Yamamoto, I.; Shiotsuka, D.; Inoue, Y.; Tokuhiro, K.; and Miwa, K. 2023. SuperDriverAI: Towards Design and Implementation for End-to-End Learning-Based Autonomous Driving. In *2023 IEEE Vehicular Networking Conference (VNC)*, 195–198. IEEE.

Bertran, M.; Tang, S.; Roth, A.; Kearns, M.; Morgenstern, J. H.; and Wu, S. Z. 2024. Scalable membership inference attacks via quantile regression. *Advances in Neural Information Processing Systems*, 36.

Carlini, N.; Chien, S.; Nasr, M.; Song, S.; Terzis, A.; and Tramer, F. 2022a. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, 1897–1914. IEEE.

Carlini, N.; Ippolito, D.; Jagielski, M.; Lee, K.; Tramer, F.; and Zhang, C. 2022b. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Carlini, N.; Liu, C.; Erlingsson, Ú.; Kos, J.; and Song, D. 2019. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, 267–284.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. Ieee.

Chatzis, S. P.; Siakoulis, V.; Petropoulos, A.; Stavroulakis, E.; and Vlachogiannakis, N. 2018. Forecasting stock market crisis events using deep and statistical machine learning techniques. *Expert systems with applications*, 112: 353–371.

Chaudhari, H.; Severi, G.; Oprea, A.; and Ullman, J. 2023. Chameleon: Increasing Label-Only Membership Leakage with Adaptive Poisoning. *arXiv preprint arXiv:2310.03838*.

Chen, J.; Jordan, M. I.; and Wainwright, M. J. 2020. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)*, 1277–1294. IEEE.

Chen, Y.; Shen, C.; Shen, Y.; Wang, C.; and Zhang, Y. 2022. Amplifying membership exposure via data poisoning. *Advances in Neural Information Processing Systems*, 35: 29830–29844.

Choquette-Choo, C. A.; Tramer, F.; Carlini, N.; and Papernot, N. 2021. Label-only membership inference attacks. In *International conference on machine learning*, 1964–1974. PMLR.

Debenedetti, E.; Severi, G.; Carlini, N.; Choquette-Choo, C. A.; Jagielski, M.; Nasr, M.; Wallace, E.; and Tramèr, F. 2024. Privacy side channels in machine learning systems. In *33rd USENIX Security Symposium (USENIX Security 24)*, 6861–6848.

Del Grosso, G.; Jalalzai, H.; Pichler, G.; Palamidessi, C.; and Piantanida, P. 2022. Leveraging adversarial examples to quantify membership information leakage. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10399–10409.

Dixit, R. R. 2021. Risk Assessment for Hospital Readmissions: Insights from Machine Learning Algorithms. *Sage Science Review of Applied Machine Learning*, 4(2): 1–15.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Guo, C.; Gardner, J.; You, Y.; Wilson, A. G.; and Weinberger, K. 2019. Simple black-box adversarial attacks. In *International conference on machine learning*, 2484–2493. PMLR.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.

Nasr, M.; Rando, J.; Carlini, N.; Hayase, J.; Jagielski, M.; Cooper, A. F.; Ippolito, D.; Choquette-Choo, C. A.; Tramèr, F.; and Lee, K. 2025. Scalable Extraction of Training Data from Aligned, Production Language Models. In *The Thirteenth International Conference on Learning Representations*.

Nasr, M.; Shokri, R.; and Houmansadr, A. 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)*, 739–753. IEEE.

Prashanth, U. S.; Deng, A.; O'Brien, K.; V, J. S.; Khan, M. A.; Borkar, J.; Choquette-Choo, C. A.; Fuehne, J. R.; Biderman, S.; Ke, T.; Lee, K.; and Saphra, N. 2025. Recite, Reconstruct, Recollect: Memorization in LMs as a Multifaceted Phenomenon. In *The Thirteenth International Conference on Learning Representations*.

Salem, A.; Zhang, Y.; Humbert, M.; Berrang, P.; Fritz, M.; and Backes, M. 2018. Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models. *arXiv preprint arXiv:1806.01246*.

Samitas, A.; Kampouris, E.; and Kenourgios, D. 2020. Machine learning as an early warning system to predict financial crisis. *International Review of Financial Analysis*, 71: 101507.

Shokri, R.; Stronati, M.; Song, C.; and Shmatikov, V. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, 3–18. IEEE.

Song, L.; and Mittal, P. 2021. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security 21)*, 2615–2632.

Song, L.; Shokri, R.; and Mittal, P. 2019. Privacy risks of securing machine learning models against adversarial examples. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, 241–257.

Tobaben, M.; Pradhan, G.; He, Y.; Jälkö, J.; and Honkela, A. 2024. Understanding Practical Membership Privacy of Deep Learning. *arXiv preprint arXiv:2402.06674*.

Tramèr, F.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*.

Tramèr, F.; Shokri, R.; San Joaquin, A.; Le, H.; Jagielski, M.; Hong, S.; and Carlini, N. 2022. Truth serum: Poisoning machine learning models to reveal their secrets. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, 2779–2792.

Watson, L.; Guo, C.; Cormode, G.; and Sablayrolles, A. 2022. On the Importance of Difficulty Calibration in Membership Inference Attacks. In *International Conference on Learning Representations*.

Yeom, S.; Giacomelli, I.; Fredrikson, M.; and Jha, S. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, 268–282. IEEE.

Yu, H.; Huo, S.; Zhu, M.; Gong, Y.; and Xiang, Y. 2024. Machine Learning-Based Vehicle Intention Trajectory Recognition and Prediction for Autonomous Driving. In *2024 7th International Conference on Advanced Algorithms and Control Engineering (ICAACE)*, 771–775. IEEE.