Purifier: Defending Data Inference Attacks via Transforming Confidence Scores

Ziqi Yang[1,2,3], Lijin Wang[1], Da Yang[1], Jie Wan[1], Ziming Zhao[1], Ee-Chien Chang[6], Fan Zhang[1,4,5*], Kui Ren[1,2,3,4]

[1] Zhejiang University
[2] ZJU-Hangzhou Global Scientific and Technological Innovation Center
[3] Key Laboratory of Blockchain and Cyberspace Governance of Zhejiang Province
[4] Jiaxing Research Institute, Zhejiang University
[5] Zhengzhou Xinda Institute of Advanced Technology, [6] National University of Singapore
{yangziqi, wanglijin, yangda, wanjie, zhaoziming, fanzhang, kuiren}@zju.edu.cn, changec@comp.nus.edu.sg

Abstract

Neural networks are susceptible to data inference attacks such as the membership inference attack, the adversarial model inversion attack and the attribute inference attack, where the attacker could infer useful information such as the membership, the reconstruction or the sensitive attributes of a data sample from the confidence scores predicted by the target classifier. In this paper, we propose a method, namely PURIFIER, to defend against membership inference attacks. It transforms the confidence score vectors predicted by the target classifier and makes purified confidence scores indistinguishable in individual shape, statistical distribution and prediction label between members and non-members. The experimental results show that PURIFIER helps defend membership inference attacks with high effectiveness and efficiency, outperforming previous defense methods, and also incurs negligible utility loss. Besides, our further experiments show that PURIFIER is also effective in defending adversarial model inversion attacks and attribute inference attacks. For example, the inversion error is raised about 4+ times on the Facescrub530 classifier, and the attribute inference accuracy drops significantly when PURIFIER is deployed in our experiment.

Introduction

Machine learning has been provided as a service by many platforms, transforming various aspects of daily life such as handling users' sensitive data. Users access these models through prediction APIs which return a confidence score or a label. Many studies have indicated that the prediction information of a sample could be exploited to perform data inference attacks to get information about this sample (Shokri et al. 2017; Yang et al. 2019; Song and Shmatikov 2020). Data inference attacks could be largely divided into two categories. The first kind of attack aims at inferring distributional information about a class by observing the prediction changes of different samples (An et al. 2022; Mehnaz et al. 2022), while the second kind of attack is to infer the individual information of a sample by observing its specific output such as the membership inference attacks (Nasr, Shokri, and Houmansadr 2018; Salem et al. 2018; Hui et al. 2021; Yeom et al. 2018; Li and Zhang 2021; Li, Li, and Ribeiro 2021), adversarial model inversion attacks (Yang et al. 2019) and attribute inference attacks (Song and Shmatikov 2020). In this paper, we focus on the second type of data inference attack.

Among these data inference attacks, *membership inference attack* (Shokri et al. 2017) is one of the most important and exemplary attacks, where the adversary is asked to determine whether a sample is in the target model's training set. Many studies acknowledge that the confidence scores tell more prediction information beyond the label and thus they should be provided in the output. Therefore, a number of approaches have been proposed to defend the membership inference attack while preserving the confidence scores (Shokri et al. 2017; Salem et al. 2018; Nasr, Shokri, and Houmansadr 2018; Abadi et al. 2016; Jia et al. 2019; Tang et al. 2022). On the other hand, some studies believe that removing the confidence information in the output is a way of defending the membership inference attack. However, these defenses are broken by label-only attacks (Yeom et al. 2018; Choquette-Choo et al. 2021; Li, Li, and Ribeiro 2021), whereby only the predicted label is exploited to infer the membership.

The major cause of membership inference attacks is that the outputs are distinguishable for members and non-members. For example, a model always behaves more confidently on predicting the training data (members) than predicting the testing data (non-members). The prediction differences between members and non-members exist in *individual shape*, *statistical distribution* and *prediction label*. (1) The target classifier often assigns a higher probability to the predicted class when given a member, making confidence scores distinguishable in individual shape. This is exploited by many attacks (Salem et al. 2018; Nasr, Shokri, and Houmansadr 2018) (2) Confidence scores in members and non-members are also distinguishable in statistical distribution. Our experiments show that confidence scores on the members are more clustered in the encoded latent space, while those on non-members are more scattered. BlindMI (Hui et al. 2021) exploits this difference to infer membership. (3) In addition, the confidence scores on members and non-members are different in prediction label. Members have a higher probability of being correctly classified than non-members, which leads to the difference in classification accuracy and is exploited by label-only attacks (Yeom et al. 2018; Li and Zhang 2021;

*Corresponding author

Li, Li, and Ribeiro 2021).

In this paper, we propose a defense method, namely PURIFIER, against the membership inference attack. The main idea is to directly reduce the distinguishability of confidence scores on members and non-members by transforming the confidence score vectors as if they were predicted on non-members. It takes as input the prediction produced by the target model and outputs a transformed version. First, we train PURIFIER on the confidence scores predicted by the target model on non-members to reconstruct these vectors using a novel training strategy. This encourages PURIFIER to learn the individual shape of these non-member confidence scores and eventually to generate confidence scores as if they were drawn from the learned pattern, reducing distinguishability of confidence scores in *individual shape*. Second, we use Conditional Variational Auto-Encoder (CVAE) as a component of PURIFIER to introduce Gaussian noises to the confidence scores, such that the statistically clustered confidence scores can be scattered and become indistinguishable from those on non-members, reducing distinguishability in *statistical distribution*. Third, to decrease the distinguishability in *prediction labels*, PURIFIER intentionally modifies the predicted labels of members while preserving those of non-members, which results in a reduction of classification accuracy gap between members and non-members.

Although PURIFIER is designed to defend the membership inference attacks, it turns out to be also effective in defending the *adversarial model inversion attack* and the *attribute inference attack*. In the adversarial model inversion attack, the adversary aims at inferring a reconstruction (Yang et al. 2019; Fredrikson, Jha, and Ristenpart 2015; Hitaj, Ateniese, and Perez-Cruz 2017) of the input. In the attribute inference attack, the adversary could infer additional attribute beyond the original input attributes of this sample (Song and Shmatikov 2020). We believe that the purification process contributes to the removal of the redundant information, and preserves only the essential information for the prediction task. As a result, the adversary can obtain no more useful information than the prediction itself from the purified prediction results.

We extensively evaluate PURIFIER on various benchmark datasets and model architectures. We empirically show that PURIFIER can defend data inference attacks effectively and efficiently with negligible utility loss. PURIFIER can reduce the membership inference accuracy. For example, the NSH attack (Nasr, Shokri, and Houmansadr 2018) accuracy drops from 70.36% to 51.71% in our experiments, which is significantly more effective than previous defenses. PURIFIER is also effective against adversarial model inversion attack. For instance, the inversion loss on the FaceScrub530 dataset is raised 4+ times (i.e. from 0.0114 to 0.0454) after applying PURIFIER. Furthermore, PURIFIER can reduce the attribute inference accuracy from 31.06% to 20.94% (almost random guessing) on one of evaluated datasets.

Contributions. In summary, we make the following contributions in this paper.

- To the best of our knowledge, our work is the first to study membership inference attacks comprehensively from the perspectives of *individual shape*, *statistical distribution* and *prediction label*.

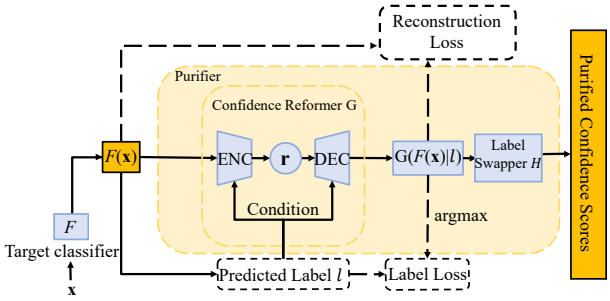

Figure 1: Architecture of PURIFIER.

- We design PURIFIER to defend against membership inference attacks by reducing the distinguishability of the confidence scores in terms of the above three aspects with negligible utility loss. PURIFIER is shown to be also effective in defending other data inference attacks.
- We extensively evaluate PURIFIER and compare it with existing defenses. Our experimental results show that PURIFIER outperforms existing defenses in both effectiveness and efficiency.

Problem Statement

We focus on classification models of neural networks, i.e., a machine learning classifier F is trained on its training dataset D_{train} to map a given sample $\mathbf{x}$ to a specific class based on the confidence vectors $F(\mathbf{x})$ which is the classifier output.

We consider the data inference attacks designed to infer useful information about a specific sample $\mathbf{x}$ based on the target classifier's output $F(\mathbf{x})$, for example, the membership inference attack, adversarial model inversion attack and attribute inference attack. We do not consider the data inference attacks (An et al. 2022; Mehnaz et al. 2022) which infer distributional information about a class through observing the output changes of F on different $\mathbf{x}$ in this paper.

$$\mathbf{x}, F, D_{aux} \rightarrow \{useful\ information\ of\ \mathbf{x}\}$$

We focus on the black-box settings, where the attacker can only query the classifier F with its data sample $\mathbf{x}$ and obtain the prediction scores $F(\mathbf{x})$. We also assume that the attacker has an auxiliary dataset D_{aux} to assist its attacks such as a set of data drawn from a similar data distribution as the target classifier's training data distribution.

Approach: PURIFIER

We propose PURIFIER as a defense against data inference attacks. Since membership inference attack is one of the most typical instances of data inference attacks, we design PURIFIER against it as the point of penetration and evaluate its defense performance against other data inference attacks. The main idea of PURIFIER is to transform the confidence score vectors in such a way that they appear indistinguishable on

members and non-members. We focus on reducing the three underlying distinguishabilities of confidence scores between members and non-members: *individual shape*, *statistical distribution* and *prediction label*.

PURIFIER consists of a *confidence reformer* G and a *label swapper* H, as shown in Figure 1. G takes as input the original confidence score vectors and reforms them as if they were predicted on non-members, achieving indistinguishability of individual shape and statistical distribution. We design G as a CVAE, with the predicted label being the condition. In this way, G is able to learn the overall distribution of the confidence scores from all classes by setting the condition to the corresponding class. The *label swapper* H takes the reformed confidence scores from G, and modifies the predicted labels of members to reduce the gap of classification accuracy between members and non-members, achieving indistinguishability of prediction label.

Achieving Individual Indistinguishability

In order to achieve the indistinguishability of individual shape between members and non-members, PURIFIER reforms the confidence scores by *confidence reformer* G, which is a CVAE. G takes the confidence score $F(\mathbf{x})$ as input, with the corresponding label l being the condition. $F(\mathbf{x})$ first goes through the encoder, where it is mapped to the encoded latent space $\mathbf{r}$. The decoder then maps the confidence score back from the latent space $\mathbf{r}$, and the reformed confidence score $G(F(x)|l)$ is obtained. G is trained on the confidence scores predicted by F on the defender's reference dataset D_{ref}, which consists of non-member samples. As a result, G learns the pattern of *individual shape* on non-members. The reforming process of G could remove difference in the individual shape of $F(\mathbf{x})$, achieving individual indistinguishability.

In order to preserve the classification accuracy, we train G to also produce the label predicted by F by adding a label loss. Formally, G is trained to minimize the following objective function.

$$L(G) = \mathop{\mathbb{E}}_{\mathbf{x}\sim p_r(\mathbf{x})}[\mathcal{R}((G(F(\mathbf{x})|l), F(\mathbf{x})) + \lambda\mathcal{L}((G(F(\mathbf{x})|l), l)] \tag{1}$$

where $p_r(\mathbf{x})$ represents the conditional probability of $\mathbf{x}$ for samples in D_{ref}, l represents the label of $F(\mathbf{x})$ (i.e., $l = \arg\max(F(\mathbf{x}))$). $\mathcal{R}$ is a reconstruction loss function (L_2 norm) and $\mathcal{L}$ is the cross entropy loss function. The parameter λ controls the balance of the two loss functions during training.

Achieving Statistical Indistinguishability

We can observe the statistical distribution of $F(\mathbf{x})$ by plotting $F(\mathbf{x})$ on the encoded latent space $\mathbf{r}$. Figure 5 shows an example of such statistical distribution on CIFAR10 dataset, where different colors represent different labels. We can observe that confidence scores are clustered into several groups according to their labels. However, the members are more clustered while non-members are not, which indicates that the distribution of members and non-members is different.

To mitigate the difference in statistical distribution between members and non-members, *confidence reformer* G introduces Gaussian noises in the latent space r, where the label l is used as the condition. During the training process, the reconstruction loss $\mathcal{R}$ encourages the decoder of G to generate confidence scores that are similar to the non-member ones on D_{ref} (non-members) with the same label l. However, noises introduced in the latent space $\mathbf{r}$ will increase the reconstruction error. As a result, G learns a robust latent representation that could preserve the statistical distribution of the non-members of label l even if noises are added. During the inference process, the added noises breakdown the clustering of confidence scores on members, while the decoder generates the reformed versions that are similar to the ones on D_{ref}, mitigating the difference in statistical distribution.

Achieving Label Indistinguishability

To cope with the difference in prediction label, we design a *label swapper* H, which modifies the prediction labels of members to reduce the gap in classification accuracy between members and non-members. After training the *confidence reformer* G, we randomly select training data to replace their predicted labels with the second largest predicted labels at a certain swap rate $p_{swap} = (acc_{train} - acc_{test})/acc_{train}$, where acc_{train} and acc_{test} are the training and the test accuracy of the target classifier respectively. Note that we fix the data at the training stage whose labels will be modified, so when attackers use the same data to query the final model, they will get the same output. Hence PURIFIER can defend the *replay attack* where attackers exploit the differences between the outputs of multiple same queries to the target model.

Given an input sample $\mathbf{x}$, H first identifies if $\mathbf{x}$ is a selected member. In order to identify members, the *label swapper* stores information of the original training data. However, it is challenging for the label swapper to efficiently store and index the member information in the run time. To this end, Label swapper stores $F(\mathbf{x})$ where $x \in D_{swap}$ as the identifiers to form a prediction indexing set P_{index} whose dimension is much smaller than the training data D_{train}. In order to tolerate small perturbations of members added by attackers to indirectly infer membership of a target member sample $x \in D_{swap}$, H uses k nearest neighbor (kNN) to identify these suspicious members and swaps their labels.

Training and Inference Process of PURIFIER

The training process of PURIFIER is detailed in Algorithm 1. For each epoch, we first draw a mini-batch of data points $\{(\mathbf{x}_{ref_j}, y_{ref})\}_{j=1}^{q}$ from the reference set D_{ref}. Then we query the target classifier F to obtain the confidence scores c_{r_j} and the labels l_{r_j} (Line 1-5). After that, the loss is calculated on the objective function 1 and gradient descent is used to update the parameters θ of *confidence reformer* G (Line 6-7). When the training of G is finished, we select the data from D_{train} at rate p_{swap} randomly to form D_{swap} (Line 10-11). After that, we query the target classifier F to get the confidence c_j of the sample $(\mathbf{x}_{train_j}, y_{train_j}) \in D_{swap}$. The original confidence score c_j is added to the prediction indexing set P_{index} and later used by the *label swapper* to achieve indistinguishability of prediction label (Line 12-15).

In the inference stage, given an input sample $\mathbf{x}$, we first

Algorithm 1: Training process of PURIFIER.

Input: The reference dataset D_{ref}, the training dataset D_{train}, the target classifier F, size of mini-batch q, size of the data need to be modify the labels t, number of epochs P, learning rate η, label loss coefficient λ

Output: Model parameters θ of *label reformer* G_θ, The prediction indexing set P_{index}

$\theta \leftarrow \mathbf{initialize}(G_\theta)$;
for $p = 1$ *to* P **do**
  for *each mini-batch* $\{(\mathbf{x}_{ref_j}, y_{ref_j})\}_{j=1}^{q} \subset D_{ref}$ **do**
    $c_{r_j} \leftarrow F(\mathbf{x}_{ref_j})$;
    $l_{r_j} \leftarrow \mathbf{onehot}(\arg\max(c_{r_j}))$;
    $g \leftarrow \nabla_\theta \frac{1}{q} \sum_{j=1}^{q} \mathcal{R}(G_\theta(c_{r_j}|l_{r_j}), c_{r_j}) + \lambda \mathcal{L}(G_\theta(c_{r_j}|l_{r_j}), l_{r_j})$;
    $\theta \leftarrow \mathbf{updateParameters}(\eta, \theta, g)$
  end
end
$P_{index} \leftarrow \varnothing$;
$D_{train} \leftarrow \mathbf{shuffle}(D_{train})$;
$D_{swap} \leftarrow \{(\mathbf{x}_{train_j}, y_{train_j})\}_{j=1}^{t} \subset D_{train}$;
for *each* $(\mathbf{x}_{train_j}, y_{train_j}) \in D_{swap}$ **do**
  $c_j \leftarrow F(\mathbf{x}_{train_j})$;
  $P_{index} \leftarrow P_{index} \cup \{c_j\}$;
end
return G_θ, P_{index}

query the target classifier F to get the confidence score c and the predicted label l. Then, we input c into the *confidence reformer* G, with l being the condition, to get the purified confidence vector p. At this stage, p is indistinguishable in individual shape and statistical distribution. The *label swapper* H checks if c has a match in P_{index} using kNN and swaps the label of p if c is matched. This ensures indistinguishability in terms of prediction label. Finally, PURIFIER returns the purified confidence scores p.

Experimental Setup

Datasets & Models

Membership inference attack. We use CIFAR10 (Shokri et al. 2017; Salem et al. 2018; Li and Zhang 2021), Purchase100 (Shokri et al. 2017; Nasr, Shokri, and Houmansadr 2018; Salem et al. 2018; Li and Zhang 2021) and FaceScrub530 (Yang et al. 2019) datasets which are widely adopted in previous studies on membership inference attacks.

Model inversion attack. We use the same datasets as membership inference attacks.

Attribute inference attack. We use the same dataset UTKFace (Zhang, Song, and Qi 2017) as in a previous study (Song and Shmatikov 2020) where the attacker infers additional attribute (i.e., race of five possible values) beyond the original gender classification task.

We attach the details of the datasets to Appendix, including the introduction, pre-processing and data allocation. We also further elucidate the target classifier and PURIFIER on different datasets in the Appendix, including their model architectures and hyper-parameters.

Existing Attacks

In our experiments, we implement the following attacks.

Membership inference attack. We implement the atttacks including ① NSH attack (Nasr, Shokri, and Houmansadr 2018), ② Mlleaks attack (Salem et al. 2018), ③ Adaptive attack (Salem et al. 2018) (where the attacker knows all the details about the defense mechanism), ④ BlindMI attack (Hui et al. 2021), ⑤ Label-only attack (Yeom et al. 2018; Li and Zhang 2021). **Model inversion attack.** The attacker uses an inversion model to reconstruct $\mathbf{x}$ according to the $F(\mathbf{x})$ (Yang et al. 2019). **Attribute inference attack.** The attacker trains a classification on D_{aux} to infer additional sensitive attribute beyond the original input attributes of the given sample (Song and Shmatikov 2020).

We attach the details of the attack above-mentioned methods, including their implementations, to the Appendix. We also attach the results of Gap attack to the Appendix.

Metrics

We use the following 4 metrics to measure the utility, defense performance and efficiency of a defense method.

①**Classification accuracy:** It is measured on the training and test set of the target classifier. ②**Inference accuracy:** This is the classification accuracy of the attacker's attack model in predicting the membership/sensitive attribute of input samples. ③**Inversion error:** Following (Yang et al. 2019), We measure the inversion error by computing the mean squared error between the original input sample and the reconstruction. ④**Efficiency:** We measure the efficiency of a defense method by reporting its training time and test time relative to the original time required by the target classifier.

Experimental Results

PURIFIER is Effective in Membership Inference

Effectiveness. Table 1 presents the defense performance of PURIFIER against different membership inference attacks. For each classification task, PURIFIER decreases the attack accuracy as well as preserves the classification accuracy. PURIFIER reduces the accuracy of NSH attack significantly for different datasets. For instance, it reduces the accuracy of NSH attack from 69.34% to 51.56% in FaceScrub530 dataset. As for Mlleaks attack, the model defended with PURIFIER reduces the attack accuracy to nearly 50%. Compared with the pure Mlleaks attack, the performance of the adaptive attack does not show a large difference where PURIFIER reduces the accuracy to nearly 50%. PURIFIER is also effective against BlindMI attack. For example, PURIFIER reduces the accuracy of BlindMI from 62.61% to 50.00% in FaceScrub530 dataset.

Comparison with other defenses. We compare PURIFIER with following defenses. ①Min-Max (Nasr, Shokri, and Houmansadr 2018). ②MemGuard (Jia et al. 2019). ③ModelStacking (Salem et al. 2018). ④MMD Defense (Li, Li, and Ribeiro 2021). ⑤SELENA (Tang et al. 2022). ⑥One-Hot Encoding. ⑦Random Noise. We attach the details to the Appendix.

Table 2 shows the defense performance of PURIFIER and other methods. PURIFIER achieves the best defense performance against most of the attacks, including the NSH attack

Dataset	Defense	Utility		Membership Inference Attack Accuracy/AUC						Inversion Error
		Train acc	Test acc.	NSH	Mlleaks	Adaptive	BlindMI	Label only attacks		L_2 norm
								Transfer	Boundary	
CIFAR10	None	99.99%	95.92%	56.03%	56.26%	N.A.	54.76%	0.5048	0.5214	1.4357
	Purifier	97.60%	95.92%	51.65%	50.26%	50.23%	50.64%	0.4974	0.4949	1.4939
Purchase100	None	100.00%	84.36%	70.36%	64.43%	N.A.	69.82%	0.5431	N.A.	0.1426
	Purifier	86.59%	83.23%	51.71%	50.09%	50.13%	50.96%	0.4978	N.A.	0.1520
FaceScrub530	None	100.00%	77.68%	69.34%	75.04%	N.A.	62.61%	0.5869	0.7739	0.0114
	Purifier	77.58%	77.52%	51.56%	51.04%	50.00%	50.00%	0.4983	0.6185	0.0454

Table 1: Defense performance of PURIFIER against various attacks. Results of Transfer attack and Boundary attack are reported in AUC. Note that the N.A. means that setting is not applicable.

Dataset	Defense	Training acc.	Test acc.	NSH Attack	Mlleaks Attack	BlindMI Attack	Inversion Error
CIFAR10	Purifier	97.60%	95.92%	**51.65%**	50.26%	**50.64%**	**1.4939**
	Min-Max	99.40%	94.38%	53.97%	52.93%	53.52%	1.4770
	MemGuard	99.99%	95.92%	53.63%	52.24%	52.03%	1.4439
	Model-Stacking	95.80%	92.12%	51.93%	51.01%	52.69%	1.4723
	MMD Defense	99.99%	87.44%	59.50%	57.60%	58.92%	1.4414
	SELENA	98.40%	93.90%	52.14%	52.35%	51.08%	1.4350
	One-Hot Encoding	99.99%	95.92%	52.17%	**50.00%**	51.88%	1.4414
	Random Noise	99.99%	95.92%	55.97%	50.01%	51.69%	1.4342
Purchase100	Purifier	86.59%	83.23%	**51.71%**	50.09%	**50.96%**	0.1520
	Min-Max	99.89%	82.03%	65.13%	63.95%	57.39%	0.1428
	MemGuard	100.00%	84.36%	62.28%	57.86%	61.35%	0.1426
	Model-Stacking	81.84%	69.68%	61.16%	55.53%	60.36%	0.1472
	MMD Defense	100.00%	82.65%	69.48%	69.89%	66.62%	0.1439
	SELENA	83.24%	79.53%	51.90%	52.97%	53.04%	0.1440
	One-Hot Encoding	100.00%	84.36%	57.65%	**50.00%**	57.67%	**0.1524**
	Random Noise	100.00%	84.36%	60.06%	50.02%	54.44%	0.1409
FaceScrub530	Purifier	77.58%	77.52%	**51.56%**	51.04%	**50.00%**	**0.0454**
	Min-Max	98.99%	68.31%	65.56%	69.84%	66.16%	0.0182
	MemGuard	100.00%	77.68%	62.48%	60.06%	62.72%	0.0117
	Model-Stacking	86.30%	57.05%	62.00%	51.86%	60.62%	0.0417
	MMD Defense	100.00%	77.38%	64.88%	67.95%	63.55%	0.0111
	SELENA	81.06%	72.05%	51.68%	51.23%	54.05%	0.0131
	One-Hot Encoding	100.00%	77.68%	57.87%	**50.00%**	61.23%	0.0420
	Random Noise	100.00%	77.68%	56.85%	50.04%	60.83%	0.0175

Table 2: Defense performance of PURIFIER and other defense methods.

and the BlindMI attack. For the Mlleaks Attack, PURIFIER can achieve the second best performance only to One-Hot Encoding and Random Noise. PURIFIER also achieves a better security-utility tradeoff. It imposes a reduction in test accuracy of about 1%. In comparison, Model-Stacking and SELENA can mitigate membership inference attacks to some extent, but they incur an intolerable reduction in model's test accuracy. For One-Hot Encoding and Random Noise, their transformation on confidence vectors leads to a large degree of semantic information loss. MemGuard can mitigate attacks with negligible decline in test accuracy. However, its defense performance is not as good as that of PURIFIER.

PURIFIER is Effective in Adversarial Model Inversion

Effectiveness. We further investigate the defense performance of PURIFIER against adversarial model inversion attack. We train an inversion attack model on top of each classifier with or without defense on FaceScrub530 dataset. Although PURIFIER is designed to protect models from membership inference attacks, it turns out that PURIFIER is also effective in mitigating model inversion attack. Figure 2 shows the results of our experiment on adversarial model inversion attack on FaceScrub530. We quantify the inversion quality by reporting the average facial similarity scores compared with the ground truth using the Microsoft Azure Face Recognition service (Microsoft 2022), which is shown on left side of Figure 2. The less the number is,the less similarity reconstructed samples share with the original samples.

We report all the inversion error under three datasets in Table 1. As shown in Table 1 and Figure 2, the inversion loss on the FaceScrub530 dataset is raised 4+ times (i.e. from 0.0114 to 0.0454) after applying PURIFIER, indicating the performance reduction of the inversion attack is significant. Note that the effect of defense against the adversarial model inversion attacks on Purchase100 and CIFAR10 seems less significant compared with FaceScrub530 This is because the inversion attack does not pe rform well on these classifiers even though without any defense.

Comparison with other defenses. PURIFIER also

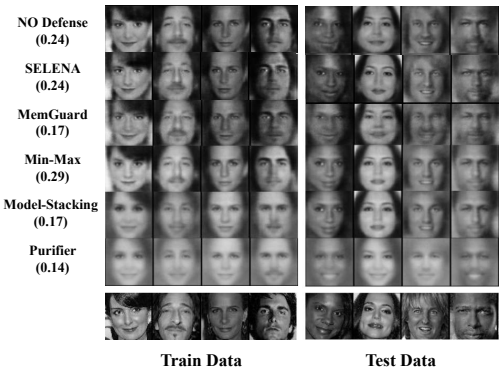

Figure 2: Model inversion attack against the FaceScrub530 classifier defended by different approaches.

Dataset	Defense	Utility		Attack Accuracy
		Train acc	Test acc	
UTKFace	None	99.92%	83.08%	31.06%
	Purifier	84.20%	82.78%	20.94%

Table 3: Attribute inference attack against the UTKFace classifier with and without PURIFIER.

achieves the best performance in defending model inversion attack on CIFAR10 and Facescrub530. Table 2 shows that PURIFIER has the largest inversion error, quantitatively demonstrating that PURIFIER achieves better defensive performance against adversarial model inversion attack than other defenses. Figure 2 depicts the reconstructed samples from confidence vectors given by each defense model on FaceScrub530 dataset. With PURIFIER as defense, reconstructed images are much less similar to the ground truth image and look more blurred. Other defense methods, however, could not protect the model from adversaries recovering small details of the original image. It can be quantitatively verified by similarity scores gathered from Microsoft Azure Face Recognition service. For instance, the average similarity score of reconstructed images of MemGuard-defended models is 0.17, which is larger than that of PURIFIER (i.e., 0.14). PURIFIER achieves the smallest similarity scores among other defense methods, indicating that PURIFIER can defend against adversarial model inversion attack effectively.

PURIFIER is Effective in Attribute Inference

Effectiveness. We deploy PURIFIER under the attribute inference attack and find that PURIFIER is also effective in mitigating it. We train an attribute inference classifier on UTKFace dataset to predict the race of the given sample. Table 3 shows the results of our experiment. The attribute inference accuracy on the UTKFace dataset is reduced to 20.94% (almost random guessing) after applying PURIFIER.

Efficiency

Figure 3 presents the efficiency of PURIFIER compared with other defenses. We attach the experimental equipment to Appendix. The training time of PURIFIER is only 0.423 times of

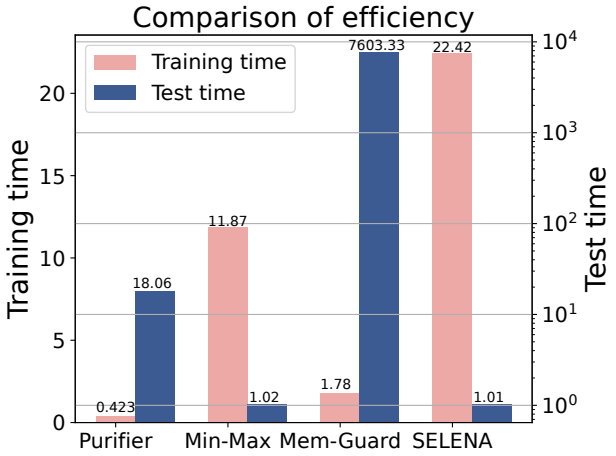

Figure 3: Efficiency of different defense methods.

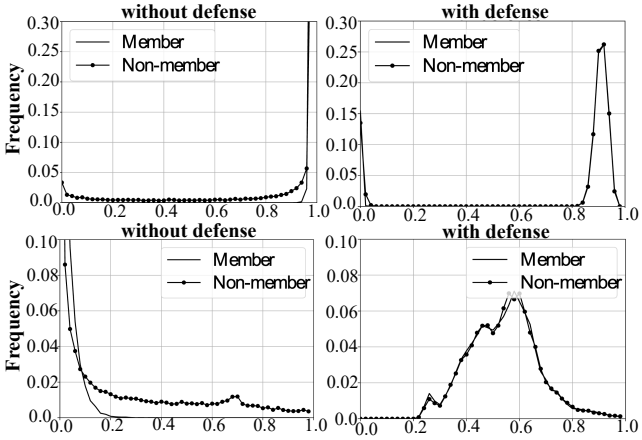

Figure 4: Distribution of the target classifier's confidence in predicting the correct class(first row) and the prediction uncertainty(second row) on members and non-members of the training set on Purchase100.

the target classifier, which outperforms all the other methods. The testing time of PURIFIER is 18.06 times as much as the target classifier, which is considered acceptable compared to MemGuard whose testing time is 7,000+ times more than the original classifier.

Analysis of Purified Confidence Scores

In this subsection, we analyze how the purified confidence scores affect membership inference attacks.

Individual indistinguishability of purified confidence. PURIFIER reshapes the confidence score vectors according to the pattern of non-members. We examine the indistinguishability of the confidence scores on members and non-members by plotting the histogram of the target classifier's confidence in predicting the correct class and the prediction uncertainty in Figure 4. The prediction uncertainty is measured as the normalized entropy $\frac{-1}{\log(k)} \sum_i \hat{\mathbf{y}}_i \log(\hat{\mathbf{y}}_i)$ of the confidence

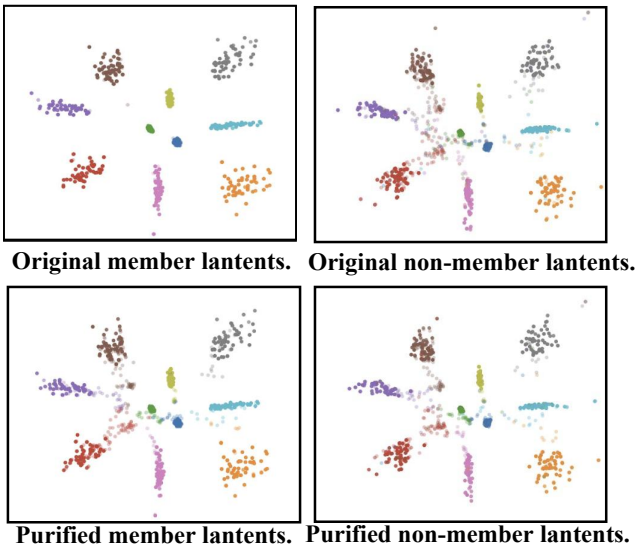

Figure 5: The statistical distribution of latent vectors in the CIFAR10 dataset. Different colors stand for latent vectors with different labels.

vector $\mathbf{y} = F(\mathbf{x})$, where k is the number of classes. As Figure 4 shows, PURIFIER can reduce the gap between the two curves. Similar results can be obtained on other classifiers and attached to the Appendix. The results of the maximum and average gaps are also listed in the Appendix. PURIFIER successfully reduces the gaps between the target classifier's confidence in predicting the correct class and the prediction uncertainty for members and non-members, which suggests it can reduce individual differences between them.

Statistical indistinguishability of purified confidence. We present the statistical distribution of confidence score vectors in the encoder latent space of the *confidence reformer*. Figure 5 visibly displays the differences in the CIFAR10 dataset between members and non-members in the latent space. As illustrated in the first row, members cluster together based on their labels while non-members are more scattered on the map. The second row of Figure 5 also shows the statistical distribution of members and non-members processed with PURIFIER in the latent space. When processed with PURIFIER, Gaussian noises are added to the confidence score vectors, making the member latent vectors to be more scattered. This demonstrates that PURIFIER can reduce the statistical differences while preserving semantic utility.

Label indistinguishability of purified confidence. PURIFIER uses *label swapper* to identify and swap the predicted labels of members. *label swapper* incurs a negligible reduction in test accuracy. At the same time, swapping the labels of the member samples reduces the training accuracy so that the gap between the accuracy of members and non-members is minimized, which is shown in Table 1. Many label-only membership inference attacks are less effective under PURIFIER with *label swapper*. This reflects that purified member confidence vectors are less distinguishable from those of the non-members in terms of the label.

Discussion

Assuming the reference data are considered as members, we present the inversion error and the inference accuracy on the reference set for each defense and attach the results to the Appendix. The Results show that PURIFIER can still preserve the defense effect against data inference attacks.

We also investigate the effect of the PURIFIER 's training data by using different in-distribution and out-of-distribution data to train PURIFIER. The results show that PURIFIER can still mitigate the attacks, but at the cost of sacrificing the utility significantly when using out-of-distribution data. We attach the results to the Appendix.

Furthermore, we investigate the effectiveness of PURIFIER to detect the members with perturbation and attach the result to Appendix. It shows that PURIFIER can accurately detect the members with perturbation $\|\eta\|_\infty < 1e-10$ on FaceScrub530 dataset.

Related Work

Data Inference Attacks

In data inference attacks, the attacker aims at inferring information about the data that the target model operates on. Xiao et al. (Xiao et al. 2019) studied the adversarial reconstruction problem. They studied the prediction model which outputs 40 binary attributes. Our paper, on the contrary, studies black-box classifiers whose output is constrained by a probability distribution. Jia and Gong (Jia and Gong 2018) proposed the adversarial formulation for privacy protection. They aimed at protecting the privacy of users' sensitive attributes from being inferred from their public data. Our work investigates inference attacks that leverage prediction results of machine learning models to infer useful information about the input data.

Secure & Privacy-Preserving Machine Learning

A number of studies made use of trusted hardware and cryptographic computing to provide secure and privacy-preserving training and the use of machine learning models. These techniques include homomorphic encryption, garbled circuits and secure multi-party computation (Liu et al. 2017; Bonawitz et al. 2017; Phong et al. 2018; Dowlin et al. 2016; Mohassel and Zhang 2017; Dwork and Feldman 2018) and secure computing using trusted hardware (Ohrimenko et al. 2016; Juvekar, Vaikuntanathan, and Chandrakasan 2018). Although these methods protect the data from direct observation by the attacker, they do not prevent information leakage via model computation.

Conclusion

In this paper, we propose PURIFIER to defend data inference attacks. PURIFIER learns the pattern of non-member confidence score vectors and purifies confidence score vectors to this pattern. It makes member confidence score vectors indistinguishable from non-members in terms of individual shape, statistical distribution and prediction label. Our experiments show that PURIFIER is effective and efficient in mitigating existing data inference attacks, outperforming previous defense methods, while imposing negligible utility loss.

Ethics Statement

The code for PURIFIER is available at https://github.com/wljLlla/Purifier_Code, and the Appendix for this paper is presented in https://arxiv.org/abs/2212.00612.

Acknowledgments

This work was supported in part by National Key R&D Program of China (2020AAA0107700), by National Natural Science Foundation of China (62102353, 62227805, 62072398), by SUTD-ZJU IDEA Grant for visiting professors (SUTD-ZJUVP201901), by National Key Laboratory of Science and Technology on Information System Security (6142111210301), by State Key Laboratory of Mathematical Engineering and Advanced Computing, and by Key Laboratory of Cyberspace Situation Awareness of Henan Province (HNTS2022001). We would like to thank Dingkun Wei, Jingjing Wang, Zijing Hu and Yanqing Liu for their implementation of some experiments.

References

Abadi, M.; Chu, A.; Goodfellow, I. J.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep Learning with Differential Privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, 308–318.

An, S.; Tao, G.; Xu, Q.; Liu, Y.; Shen, G.; Yao, Y.; Xu, J.; and Zhang, X. 2022. MIRROR: Model Inversion for Deep Learning Network with High Fidelity. In *Proceedings of the Network and Distributed Systems Security Symposium (NDSS 2022)*.

Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H. B.; Patel, S.; Ramage, D.; Segal, A.; and Seth, K. 2017. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security - CCS '17*, 1175–1191. ISBN 978-1-4503-4946-8.

Choquette-Choo, C. A.; Tramer, F.; Carlini, N.; and Papernot, N. 2021. Label-Only Membership Inference Attacks. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 1964–1974. PMLR.

Dowlin, N.; Gilad-Bachrach, R.; Laine, K.; Lauter, K.; Naehrig, M.; and Wernsing, J. 2016. CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, 201–210. JMLR.org. Event-place: New York, NY, USA.

Dwork, C.; and Feldman, V. 2018. Privacy-Preserving Prediction. In Bubeck, S.; Perchet, V.; and Rigollet, P., eds., *Proceedings of the 31st Conference On Learning Theory*, volume 75 of *Proceedings of Machine Learning Research*, 1693–1702. PMLR.

Fredrikson, M.; Jha, S.; and Ristenpart, T. 2015. Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*, 1322–1333.

Hitaj, B.; Ateniese, G.; and Perez-Cruz, F. 2017. Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, volume 1, 603–618. ISBN 978-1-4503-4946-8.

Hui, B.; Yang, Y.; Yuan, H.; Burlina, P.; Gong, N. Z.; and Cao, Y. 2021. Practical blind membership inference attack via differential comparisons. *arXiv preprint arXiv:2101.01341*.

Jia, J.; and Gong, N. Z. 2018. AttriGuard: A Practical Defense Against Attribute Inference Attacks via Adversarial Machine Learning. In *27th USENIX Security Symposium (USENIX Security 18)*, 513–529. {USENIX} Association. ISBN 978-1-931971-46-1. Event-place: Baltimore, MD.

Jia, J.; Salem, A.; Backes, M.; Zhang, Y.; and Gong, N. Z. 2019. MemGuard: Defending against Black-Box Membership Inference Attacks via Adversarial Examples. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security - CCS '19*, 259–274. ACM Press. ISBN 978-1-4503-6747-9. Event-place: London, United Kingdom.

Juvekar, C.; Vaikuntanathan, V.; and Chandrakasan, A. 2018. Gazelle: A Low Latency Framework for Secure Neural Network Inference. In *27th USENIX Security Symposium (USENIX Security 18)*. ISBN 978-1-931971-46-1.

Li, J.; Li, N.; and Ribeiro, B. 2021. Membership Inference Attacks and Defenses in Classification Models. In *Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy*, CODASPY '21, 5–16. Association for Computing Machinery. ISBN 978-1-4503-8143-7.

Li, Z.; and Zhang, Y. 2021. Membership leakage in label-only exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 880–895.

Liu, J.; Juuti, M.; Lu, Y.; and Asokan, N. 2017. Oblivious Neural Network Predictions via MiniONN Transformations. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, 2017, 619–631.

Mehnaz, S.; Dibbo, S. V.; Kabir, E.; Li, N.; and Bertino, E. 2022. Are Your Sensitive Attributes Private? Novel Model Inversion Attribute Inference Attacks on Classification Models. In *31st USENIX Security Symposium (USENIX Security 22)*, 4579–4596. Boston, MA: USENIX Association. ISBN 978-1-939133-31-1.

Microsoft. 2022. Azure. https://azure.microsoft.com/en-us/services/cognitive-services/face/. Accessed: 2022-6-6.

Mohassel, P.; and Zhang, Y. 2017. SecureML: A System for Scalable Privacy-Preserving Machine Learning. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, 19–38.

Nasr, M.; Shokri, R.; and Houmansadr, A. 2018. Machine Learning with Membership Privacy Using Adversarial Regularization. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, 634–646. ACM. ISBN 978-1-4503-5693-0. Event-place: Toronto, Canada.

Ohrimenko, O.; Schuster, F.; Fournet, C.; Mehta, A.; Nowozin, S.; Vaswani, K.; and Costa, M. 2016. Oblivious Multi-Party Machine Learning on Trusted Processors. In *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016.*, 619–636.

Phong, L. T.; Aono, Y.; Hayashi, T.; Wang, L.; and Moriai, S. 2018. Privacy-Preserving Deep Learning via Additively Homomorphic Encryption. 13(5): 1333–1345.

Salem, A.; Zhang, Y.; Humbert, M.; Fritz, M.; and Backes, M. 2018. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS 2019)*.

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 2017, San Jose, CA, USA, May 22-26, 2017*, 3–18.

Song, C.; and Shmatikov, V. 2020. Overlearning Reveals Sensitive Attributes. In *8th International Conference on Learning Representations, ICLR 2020*.

Tang, X.; Mahloujifar, S.; Song, L.; Shejwalkar, V.; Nasr, M.; Houmansadr, A.; and Mittal, P. 2022. Mitigating Membership Inference Attacks by Self-Distillation Through a Novel Ensemble Architecture. In *31st USENIX Security Symposium (USENIX Security 22)*, 1433–1450. Boston, MA: USENIX Association. ISBN 978-1-939133-31-1.

Xiao, T.; Tsai, Y.-H.; Sohn, K.; Chandraker, M.; and Yang, M.-H. 2019. Adversarial Learning of Privacy-Preserving and Task-Oriented Representations.

Yang, Z.; Zhang, J.; Chang, E.-C.; and Liang, Z. 2019. Neural Network Inversion in Adversarial Setting via Background Knowledge Alignment. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, 225–240. Association for Computing Machinery. ISBN 978-1-4503-6747-9. Event-place: New York, NY, USA.

Yeom, S.; Giacomelli, I.; Fredrikson, M.; and Jha, S. 2018. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting. 268–282.

Zhang, Z.; Song, Y.; and Qi, H. 2017. Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5810–5818.