SAME: Sample Reconstruction against Model Extraction Attacks

Yi Xie[1*], Jie Zhang[2], Shiqian Zhao[2], Tianwei Zhang[2], Xiaofeng Chen[1†]

[1]Xidian University, China
[2]Nanyang Technological University, Singapore
xieyi@stu.xidian.edu.cn, {jie_zhang, shiqian.zhao, tianwei.zhang}@ntu.edu.sg, xfchen@xidian.edu.cn

Abstract

While deep learning models have shown significant performance across various domains, their deployment needs extensive resources and advanced computing infrastructure. As a solution, Machine Learning as a Service (MLaaS) has emerged, lowering the barriers for users to release or productize their deep learning models. However, previous studies have highlighted potential privacy and security concerns associated with MLaaS, and one primary threat is model extraction attacks. To address this, there are many defense solutions but they suffer from unrealistic assumptions and generalization issues, making them less practical for reliable protection. Driven by these limitations, we introduce a novel defense mechanism, SAME, based on the concept of sample reconstruction. This strategy imposes minimal prerequisites on the defender's capabilities, eliminating the need for auxiliary Out-of-Distribution (OOD) datasets, user query history, white-box model access, and additional intervention during model training. It is compatible with existing active defense methods. Our extensive experiments corroborate the superior efficacy of SAME over state-of-the-art solutions. Our code is available at https://github.com/xythink/SAME.

Introduction

Deep learning models have demonstrated superior performance in various domains. Yet, they often demand significant resources, including vast training data, advanced computational capabilities, and rigorous parameter optimization efforts. These requirements make deep learning models invaluable and expensive for adoption. Consequently, Machine Learning as a Service (MLaaS) has garnered significant interest, offering users a simplified and cost-efficient avenue to deploy sophisticated models.

Despite these advantages, a significant body of research has also revealed the privacy and security risks of models deployed with MLaaS (Tramèr et al. 2016; Shokri et al. 2017; Liu et al. 2021; Yang et al. 2023; Lou et al. 2021). Among these, model-extraction attacks (Yu et al. 2020; Pal et al. 2020; Zhao et al. 2023; Chen et al. 2021; Li et al. 2022) represent a prominent threat, posing a direct risk to the intellectual property rights of the model owner. The objective of model-extraction attacks is to locally recreate the target model at a minimal cost, leveraging limited queries to the openly deployed victim model. To achieve this goal, earlier works have proposed different strategies to generate the query samples for efficient model stealing, including surrogate sample (Pal et al. 2020; Orekondy, Schiele, and Fritz 2019), adversarial sample (Yu et al. 2020; Papernot et al. 2017), and synthetic sample (Barbalau et al. 2020; Truong et al. 2021; Kariyappa, Prakash, and Qureshi 2021). These attacks exhibit high effectiveness and efficiency across different threat environments, deep learning models, and tasks.

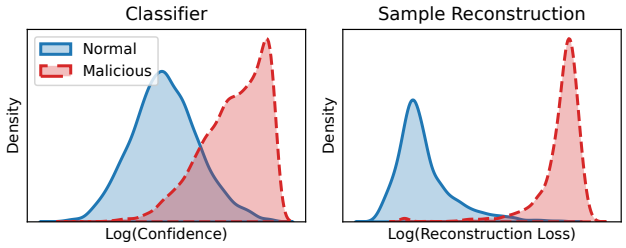

Figure 1: Distributions of anomaly scores for the classifier-based detection (left) and our sample reconstruction-based detection (right). The x-axis is in the logarithmic scale due to its long-tailed distribution. We utilize MNIST as normal query samples and employ KnockoffNets (with EMNIST-digits as the proxy set) to generate the malicious query samples. All samples undergo consistent preprocessing.

Many efforts have been made to mitigate model extraction attacks (Jiang et al. 2023). Among them, malicious sample detection is the mainstream strategy. The model owner aims to distinguish the query samples used for model extraction from normal ones, and then reject them or return obfuscated responses. However, the advance and diversity of attack approaches pose several challenges in designing an effective detector. **First**, some defenses, such as Prada (Juuti et al. 2019), require keeping a record of each user's queries for anomaly detection. They become vulnerable when the adversary launches a distributed attack (Yao et al. 2023). **Second**, some approaches build machine learning classifiers to differentiate malicious and normal samples based on their features or predicted confidence scores (Kariyappa and

*This work was done at NTU as a visiting student.
†Corresponding author

Qureshi 2020). They are less effective in handling out-of-distribution (OOD) samples, even if the *Outlier Exposure* (OE) strategy (Hendrycks, Mazeika, and Dietterich 2018) is adopted. As shown in Figure 1 (left), the malicious samples have a large overlap with normal ones, indicating that lots of samples will be misclassified. **Third**, some methods leverage *Ensemble of Diverse Models* (EDM) to detect malicious samples. They necessitate training several duplicates of the victim model using both in-distribution and auxiliary OOD samples, which significantly amplifies the defense costs.

To address these challenges, we propose a novel detection method: Sample reconstruction Against Model Extraction (SAME). Our observation is that in-distribution and out-of-distribution samples exhibit significantly different features in the reconstruction process, as shown in Figure 1 (right). This inspires us to leverage sample reconstruction to detect malicious queries. Specifically, SAME adopts the Masked Auto-encoder (He et al. 2022) to reconstruct each query sample. It further builds an Auxiliary Model to repair the prediction of the reconstructed sample. Finally, it judges suspicious queries by combining two sources of information: reconstruction loss from the Masked Auto-encoder and deviation loss from the Auxiliary Model. Compared to existing works, SAME imposes minimal prerequisites on the defender's capabilities: (1) it eliminates the need for an auxiliary OOD training dataset; (2) it avoids retaining the user's query history; (3) it removes the demand for white-box access to the victim model. We conduct extensive experiments to demonstrate the superiority of SAME in detecting different types of model extraction attacks over SOTA methods.

In summary, the main contribution of this paper includes three aspects:

- We reveal the inherent weaknesses of classifier-based detection mechanisms, especially when confronted with unseen malicious queries in model extraction scenarios.
- We introduce SAME, a novel malicious query detection method rooted in sample reconstruction, significantly reducing the demands on defenders and acting as a versatile add-on to bolster current active defense strategies.
- We demonstrate the effectiveness of SAME under multiple attack types through extensive experiments.

Preliminaries

In this section, we first introduce model extraction attack (MEA) and the corresponding detection methods. Then, the threat model is provided in detail.

Model Extraction Attack

Given a victim model F_V (typically considered a black-box), the objective of a Model Extraction Attack (MEA) is to derive a functionally equivalent substitute model F_S for illegal purposes (e.g., intellectual property violation). This could be formulated as minimizing the similarity loss on the victim model test set D_V^{test}:

$$\min_{F_S} \sum_{x \in D_V^{test}} \mathcal{L}(F_V(x), F_S(x)), \tag{1}$$

where x denotes the samples and $\mathcal{L}$ is a loss function measuring the discrepancy between the outputs of F_V and F_S.

The performance of the substitute model is highly affected by the query samples submitted by the attacker (Orekondy, Schiele, and Fritz 2019). Past works have proposed different methodologies to construct query samples to improve extraction accuracy and efficiency. They can be classified into the following three categories.

Sampling-based Stealing. This type of attack aims to construct a query dataset from a proxy dataset (often composed of public datasets) using a sampling strategy. Since different samples can provide different amounts of information to the substitute model, an appropriate sampling strategy can improve the attack performance. Knockoff (Orekondy, Schiele, and Fritz 2019) likens the sampling strategy to a multi-armed bandit problem in reinforcement learning. It adjusts the sampling strategy for the next step according to the reward from the previous actions. In addition, a series of works (Pal et al. 2020; Chandrasekaran et al. 2020) use active learning to improve the stealing efficiency.

Perturbation-based Stealing. It was pointed out that samples lying approximately on the decision boundary of the victim model can greatly reduce the query cost (Yu et al. 2020; Wang et al. 2021). Therefore some works introduce the perturbation-based strategy to generate query samples distributed near the decision boundary. As a representative, JBDA (Papernot et al. 2017) proposes perturbation based on the Jacobian matrix on a small number of original victim training samples. CloudLeak (Yu et al. 2020) uses a variety of adversarial perturbation methods to generate samples that approximate the model's decision boundary. Extensive experiments demonstrate the benefit of these perturbation strategies in boosting the stealing performance.

Synthetic-based Stealing. In most scenarios, the adversary does not have any dataset for model extraction. He can only generate noise samples for stealing (Truong et al. 2021). The mainstream strategy is to use the gradient approximation method (Truong et al. 2021; Kariyappa, Prakash, and Qureshi 2021) to generate query samples, which can obtain more information regarding the victim model.

Model Extraction Attack Detection

A popular defense direction is to detect the malicious query samples used for model extraction. This can be formulated as a binary classification problem. For each query, the model owner determines whether the sample is from OOD or not by calculating an anomaly score $S(x)$, and comparing it with a threshold λ. A higher $S(x)$ indicates a greater possibility that this sample x is from OOD. It is important to minimize the misclassification of samples from ID (normal queries).

There are different strategies for building such an anomaly detector. One possible direction is to detect the extraction activity at the **user level**. For instance, Prada (Juuti et al. 2019) keeps query logs for all users to spot potential suspicious activities. However, as pointed out by some works (Yao et al. 2023), user-level detection based on the query history cannot mitigate distributed attacks, where the adversary employs multiple accounts to query the victim model. Hence, a more promising direction is **sample-level**

defense (Kariyappa and Qureshi 2020; Kariyappa, Prakash, and Qureshi 2020; Dziedzic et al. 2021), which performs detection on each sample. Existing solutions can be mainly divided into the following two types:

Outlier Exposure (OE). In practical settings, it is implausible for the model owner to have prior knowledge of the malicious query set $\mathcal{D}_A$. Therefore, some works (Kariyappa and Qureshi 2020) introduce an auxiliary outlier set $\mathcal{D}_{OE}$, which is disjoint from $\mathcal{D}_A$, to assist in learning a classifier for potential outliers. By exposing $\mathcal{D}_{OE}$ to F_V during training, OE makes F_V produce uniform probability distribution $\mathcal{U}$ on outliers. The optimization equation is:

$$\mathbb{E}_{x\sim\mathcal{D}_V}\left[\mathcal{L}(F_V(x),y)\right]+\gamma\mathbb{E}_{x'\sim\mathcal{D}_{OE}}\left[\mathcal{L}_{OE}(F_V(x'),\mathcal{U})\right], \tag{2}$$

where $\mathcal{L}$ is the original learning objective, and $\mathcal{L}_{OE}$ is the outlier exposure loss.

However, OE needs to incorporate an auxiliary dataset into the training process of the victim model, which will introduce additional training overhead. Furthermore, additional learning objectives can degrade the accuracy of the model on the original task (see Table 3).

Ensemble of Diverse Models (EDM). EDM utilizes an ensemble of diverse models $\{f_i\}_{i=1}^{i=N}$ to produce discontinuous predictions for OOD data (Kariyappa, Prakash, and Qureshi 2020). Similar to OE, EDM also leverages an auxiliary outlier set $\mathcal{D}_{OE}$ to defend against model stealing. Specifically, $\{f_i\}_{i=1}^{i=N}$ are trained jointly on $\mathcal{D}_V^{train}$ and $\mathcal{D}_{OE}$ according to the accuracy and diversity objectives:

$$\mathcal{L}=\mathbb{E}_{x\sim\mathcal{D}_V,x'\sim\mathcal{D}_{OE}}\left[(\frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(f_i(x),y))+\gamma\mathcal{L}_{Div}(f_i(x'))\right], \tag{3}$$

where the first loss term ensures the model utility on $\mathcal{D}_V$, and the second term ensures the diversity of predictions for a single outlier sample form $\mathcal{D}_{OE}$ across multiple submodels. Since this method has no explicit anomaly score, we compute the score based on the consensus among these diverse models, and it is smaller when models agree. This idea is also used in previous work (Dziedzic et al. 2021).

We shall point out that the effectiveness of detection based on auxiliary OOD datasets largely depends on the similarity between the distributions of auxiliary datasets and real malicious queries. The detector will perform poorly when such a distribution gap is large.

Threat Model

Attacker's Ability and Goal. As mentioned above, the goal of the attacker is to obtain his substitute model F_S, which is functionally similar to the victim model F_V. To achieve it, we assume that the attacker can leverage any MEA strategies. In this paper, we adopt three popular MEAs for implementation, namely, KnockoffNets (Knockoff) (Orekondy, Schiele, and Fritz 2019), Jacobian-Based Dataset Augmentation (JBDA) (Papernot et al. 2017), and Data-Free Model Extraction (DFME) (Truong et al. 2021). Additionally, malicious queries submitted by attackers may vary in proportion to the overall queries (see Table 4).

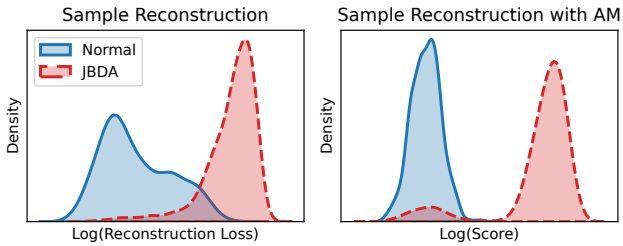

Figure 2: Distributions of anomaly scores for the reconstruction-based detection without (left) and with (right) Auxiliary Model (AM). The x-axis is in the logarithmic scale due to its long-tailed distribution. We utilize CIFAR-10 as normal query samples and employ JBDA (with 200 seed samples) to generate the malicious query samples. All samples undergo consistent preprocessing.

Defender's Ability and Goal. In this paper, we mainly focus on defending MEA by detection methods. In other words, the defender shall detect the malicious query fed by the attacker. Contrary to existing MEA detection methods (Kariyappa and Qureshi 2020; Kariyappa, Prakash, and Qureshi 2020), we consider a more practical scenario as follows: 1) the defender does not need auxiliary OOD datasets or users' query history, and he cannot interfere with the victim model training process; 2) more importantly, the defender is unaware of the distribution of malicious queries under different attack strategies, namely, the defense is expected to be general over different types of model extraction.

In a nutshell, the ability of the defender is more limited, inducing greater challenges for our MEA detection.

Motivation

The design of SAME is motivated by two observations.

First, **sample reconstruction can better disclose anomaly than sample classification.** Existing detection methods build DNN classifiers to detect suspicious samples. However, numerous studies have shown that even with the Outlier Exposure (OE) strategy, OOD detection can exhibit over-confidence in unseen OOD samples (Nguyen, Yosinski, and Clune 2015; Li et al. 2023). As depicted in Figure 1 (left), the introduction of the OE strategy cannot effectively differentiate the majority of malicious queries from benign ones. Instead of directly building the detection classifier, our key insight is to reconstruct the query samples and identify the anomalies from the reconstruction process. This is based on the observation that the reconstruction loss is a better indicator of malicious samples than classifier-based confidence scores in the model extraction scenario. For the first time, we introduce the idea of autoencoder-based sample reconstruction for model extraction attack detection.

Second, **an auxiliary model can better facilitate the OOD detection.** We find that sample reconstruction is effective for sample-based stealing and synthetic-based stealing, but not perturbation-based stealing, wherein the attacker has a small number of original datasets, and the whole query set is perturbed on these seed samples. In other words, the

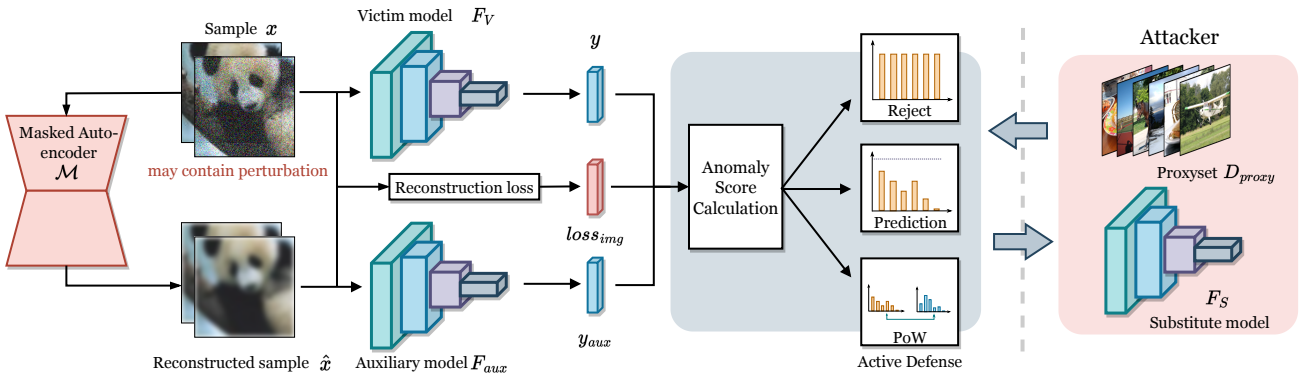

Figure 3: The workflow of the proposed SAME. Whenever a sample is received, a fully trained masked auto-encoder first performs sample reconstruction. The reconstructed sample is then fed into an auxiliary model that outputs an auxiliary prediction. The overall anomaly score is calculated based on two samples and two predictions. After that, an appropriate response strategy is selected according to the anomaly score. The victim model remains frozen throughout the defense.

reconstruction loss for perturbation-based queries is similar to the one for benign queries, as shown in Figure 2 (left). To address it, we further introduce an auxiliary model to distinguish such perturbation-based queries. Specifically, we adopt a copy of the original victim model as the architecture of the auxiliary model but train it on a reconstructed version (using MAE) of the original dataset. In this way, for perturbation-based queries, we will obtain different predictions from the victim model and the auxiliary model. But for the other queries, the predictions tend to be consistent. Based on this, we can easily detect malicious queries by perturbation-based stealing, as shown in Figure 2 (right).

Methodology

Based on the above analysis, we introduce SAME, a novel model extraction attack detection methodology. It uses a sample reconstruction strategy based on Masked Auto-encoder to disclose the malicious behaviors of query samples. It further integrates an Auxiliary Model to repair the model prediction and reinforce the detection results. Compared to prior works, SAME can minimize the requirements for the defender's capabilities: i.e., he does not need extra datasets or white-box access to the victim model. Figure 3 shows the overview of SAME, which consists of three stages: (1) Sample Reconstruction via Masked Auto-encoder; (2) Attack Repairing via Auxiliary Model; (3) Anomaly Score Calculation. We will explain each stage in the following part.

Sample Reconstruction via Masked Auto-encoder

The first stage of our detection pipeline is reconstructing the query sample with an auto-encoder. Auto-encoders, as unsupervised neural network architectures, are primarily employed for dimensionality reduction and feature learning. Despite their effectiveness in many scenarios, traditional auto-encoders might not always capture the most salient features, especially when dealing with noisy datasets. To address this issue, we employ a *masked autoencoder* (MAE) for sample reconstruction. A masked autoencoder introduces an additional masking operation during the encoding phase. Before feeding the input data to the encoder, a mask is applied, forcing the encoder to focus only on specific portions of the data. The mask essentially provides a form of inductive bias, directing the model to concentrate on potentially informative segments of the input.

In SAME, the masked autoencoder $\mathcal{M}$ consists of an encoder $f_\theta : \mathbb{R}^n \to \mathbb{R}^m$ and a decoder $g_\phi : \mathbb{R}^m \to \mathbb{R}^n$, parameterized as θ and ϕ respectively. Let $x \in D_V$ be a sample in the victim training set D_V, and $b \in \{0,1\}^n$ be the mask matrix that is sampled following a probability distribution. During the training process, each sample x will be masked and then passed to the encoder to get the latent variable $z = f_\theta(b \odot x)$. Afterward, the latent variable will be passed to the decoder g_ϕ to get the reconstructed sample $\hat{x} = \mathcal{M}(x) = (g_\phi \circ f_\theta)(b \odot x)$. The objective function of $\mathcal{M}$ is to compute the following MSE loss between the original input x and reconstructed sample $\hat{x}$:

$$L_{MAE} = \frac{1}{|D_V|} \sum_{x \in D_V} \|x - \hat{x}\|^2, \tag{4}$$

where the loss is minimized when the model is fully trained on the victim dataset. Afterward, $\mathcal{M}$ is used for score calculation and attack repair in the subsequent steps.

Attack Repair via Auxiliary Model

We denote the Auxiliary Model as F_{aux}, to repair and reinforce the anomaly detection. Given our original dataset D_V, the masked autoencoder $\mathcal{M}$ processes each sample to produce a reconstructed dataset $\hat{D_V}$, where every sample $\hat{x_i} \in \hat{D_V}$ corresponds to a sample $x_i \in D_V$. We then train F_{aux} using the reconstructed samples $\hat{X}$, paired with their respective original labels $Y = \{y_1, y_2, ..., y_N\}$. The goal here is to ensure that the predictions of F_{aux} on $\hat{D_V}$ align as closely as possible with the predictions of the original victim model on D_V. Thus, the objective function for training

F_{aux} can be defined as:

$$L_{aux} = \frac{1}{N}\sum_{i=1}^{N}\|F_V(x_i) - F_{aux}(\hat{x}_i)\|^2, \quad (5)$$

where F_V denotes the victim model, and the objective is to minimize the squared difference between the predictions of F_V and F_{aux} across all samples. By achieving this, we aim to ensure that the Auxiliary Model is a faithful reflection of the victim model's behavior but operates in the transformed space of the reconstructed dataset $\hat{D_V}$.

Anomaly Score Calculation

The primary goal of SAME is to output an anomaly score that indicates the malicious level of the query sample. The score consists of two parts: (1) the reconstruction loss from the masked auto-encoder; (2) the deviation loss from the auxiliary model. The whole score could be calculated as follows:

$$S(x) = \alpha \cdot \|x - \hat{x}\|^2 + (1-\alpha) \cdot \|F_V(x) - F_{aux}(\hat{x})\|^2, \quad (6)$$

where α is a hyperparameter to balance the two score items. We perform ablation studies in the next section to evaluate the impact of this hyperparameter.

Flexibility as an Add-on. After obtaining the anomaly score, the defender can choose subsequent active defense strategies to weaken the adversary's performance. Without loss of generality, we implement two common defense strategies: reject prediction and proof-of-work. The former rejects responding to queries with high anomaly scores, which can prevent the attacker from obtaining sensitive information. The latter requires users to complete a proof-of-work (PoW) before they can get the prediction. And the difficulty of the PoW problem is tied to the anomaly scores.

Experiment

Experimental Settings

Datasets and Model Architectures. We evaluate our scheme on two groups of datasets: 1) MNIST (LeCun et al. 1998) and EMNIST-digits (Cohen et al. 2017); 2) CIFAR-10 and CIFAR-100 (Krizhevsky, Hinton et al. 2009). Specifically, the victim model is trained on MNIST and CIFAR-10, while EMNIST-digits and CIFAR-100 serve as datasets for the attacker. Since the comparison scheme requires auxiliary anomaly datasets, we use KMNIST (Clanuwat et al. 2018) for MNIST and Tiny ImageNet (Le and Yang 2015) for CIFAR-10. To evaluate the performance of the defense in extreme cases, the attacker adopts the same model structure for his substitute model as the victim model: Conv3 (three-layer CNN) (LeCun et al. 1989) for MNIST and ResNet-18 for CIFAR-10. In all experiments, we use a MAE model based on the ViT-Tiny encoder (Dosovitskiy et al. 2020), trained for 500 epochs on the victim training set.

Attack Methods. We use three different attack methods: (1) KnockoffNets (Knockoff): as a sampling-based attack, it uses reinforcement learning to choose samples from the proxy dataset. (2) Jacobian-Based Dataset Augmentation (JBDA): this method uses a Jacobian-based data augmentation algorithm to generate new samples from seed samples. We utilize a seed dataset comprising 200 images, with a perturbation step size λ set to 0.1. (3) Data-Free Model Extraction (DFME): this method belongs to the synthesis-based category. The attacker does not need any proxy dataset, which will lead to a decrease in attack performance.

Baseline Methods. We choose two SOTA defense solutions as discussed above: (1) Outlier Exposure (OE); (2) Ensemble-based defense (EDM). Both methods need the model defender to collect an auxiliary malicious dataset. Then the victim model is trained in an adversarial manner. In contrast, our method does not require any auxiliary datasets, which is a more practical assumption.

Metrics. To quantitatively evaluate the performance, we adopt three metrics: Area Under the Receiver Operating Characteristic curve (AUROC), Area Under the Precision-Recall curve (AUPR), and the False Positive Rate at N% true positive rate (FPRN). AUROC evaluates the overall performance, while AUPR focuses on precision and recall particularly in imbalanced datasets. Besides, FPRN can better measure the trade-off between sensitivity and specificity.

Comparisons with the Baseline Methods

Effectiveness. Across all attack methods and datasets, SAME demonstrates generally superior performance compared to the baseline methods OE and EDM, particularly in AUROC and AUPR, as shown in Tables 1 and 2. In MNIST, SAME consistently outperforms the baseline methods, especially against JBDA and Knockoff attacks. For the result under the 1k query budget, SAME achieves an impressive AUROC score of 93.30% and 99.37% for JBDA and Knockoff respectively. Similarly, AUPR scores are significantly higher for SAME, especially compared to OE and EDM. Increasing the query budget to 4k does not lead to a substantial difference in the performance metrics for SAME, suggesting its robustness irrespective of the attack cost.

The superiority of SAME is more pronounced on the CIFAR10 dataset, further emphasizing its strength. For instance, with a 6k query budget under the Knockoff attack, SAME achieves an AUROC of 92.53% which is significantly higher than that of OE (77.38%) and EDM (71.72%). For the DFME attack, the performance of SAME reaches a remarkable 100.00% in AUROC and AUPR under the 6k query budget, a feat unmatched by the other defenses. Interestingly, for the 10k query budget, SAME still retains its lead, especially against the Knockoff and DFME attacks.

SAME manages to achieve the lowest FPR95 for most attacks and settings, which is essential for practical implementations. A low FPR ensures that normal queries are not mistakenly classified as malicious, which otherwise could interrupt the normal service or degrade the users' experience.

In summary, SAME demonstrates robustness and superiority against various attacks across both datasets, highlighting its potential as a reliable defense mechanism.

Fidelity and Efficiency. We also evaluate the impact of different defense methods on the performance of the victim

B	Method	D_{OE}	JBDA			Knockoff			DFME		
			AUROC↑	AUPR↑	FPR95↓	AUROC↑	AUPR↑	FPR95↓	AUROC↑	AUPR↑	FPR95↓
1k	OE	KMNIST	58.70	56.28	84.60	91.78	92.59	43.60	100.00	100.00	0.00
	EDM	KMNIST	66.06	59.73	79.60	86.01	86.70	64.20	100.00	100.00	0.00
	SAME	-	**93.30**	**92.95**	**27.40**	**99.37**	**99.47**	**1.30**	100.00	100.00	0.00
4k	OE	KMNIST	60.05	56.79	83.50	91.70	92.24	42.90	**100.00**	**100.00**	0.03
	EDM	KMNIST	71.37	65.42	74.65	85.04	85.63	67.11	99.99	99.99	**0.00**
	SAME	-	**93.10**	**92.84**	**28.52**	**99.19**	**99.38**	**1.25**	99.91	99.75	0.15

Table 1: AUROC (%), AUPR (%), and FPR95 (%) of different detection methods under three different attacks on MNIST.

B	Method	D_{OE}	JBDA			Knockoff			DFME		
			AUROC↑	AUPR↑	FPR95↓	AUROC↑	AUPR↑	FPR95↓	AUROC↑	AUPR↑	FPR95↓
6k	OE	ImageNet-T	74.53	70.41	64.37	77.38	72.74	60.77	74.02	60.64	44.30
	EDM	ImageNet-T	73.59	69.50	71.35	71.72	67.63	72.52	80.35	73.55	49.72
	SAME	-	**80.89**	**79.07**	**56.68**	**84.03**	**79.97**	**43.98**	**97.64**	**97.43**	**12.52**
10k	OE	ImageNet-T	73.68	69.53	63.14	77.82	72.96	59.96	76.17	62.99	43.28
	EDM	ImageNet-T	74.17	69.89	69.98	71.27	67.02	73.03	81.41	75.31	48.52
	SAME	-	**80.66**	**78.77**	**58.18**	**83.84**	**79.71**	**44.13**	**98.21**	**97.96**	**8.02**

Table 2: AUROC (%), AUPR (%), and FPR95 (%) of different detection methods under three different attacks on CIFAR-10.

Method	Model Accuracy (%)↑	Training Time (s)↓
OE	80.83	1562.92
EDM	80.37	2014.80
SAME	**83.28**	**170.44**

Table 3: Accuracy and training time of the victim model under OE, EDM, and SAME.

model, as shown in Table 3. It demonstrates the negative impact of the OE and EDM strategies on the victim model in accuracy degradation and long training time. Moreover, it is observed that the implementation of a structure-sharing policy between the victim model and the auxiliary model can decrease memory usage without harming victim model accuracy. In summary, SAME is more suitable for malicious query detection in the model-stealing scenario.

Flexibility as an Add-on. We test the performance of integrating SAME with two active defense methods: reject prediction and proof-of-work. Among them, reject prediction can be regarded as a special case of the perturbation-based defense. We use two metrics to evaluate the defense performance after splicing: the number of successful queries and response time, for both normal and malicious queries. As shown in Figure 4, when using SAME as a detection plug-in, queries submitted by normal users are least negatively affected, while malicious queries are largely disturbed, on both metrics. This reflects the effectiveness and compatibility of SAME as a detection strategy.

Ablation Study

Effects of Different Components. To demonstrate the effectiveness of each stage in SAME, we evaluate the performance of two variants of SAME: (1) SAME−X: only keeping the loss term based on sample reconstruction; (2) SAME−Y: only keeping the deviation loss item based on the Auxiliary

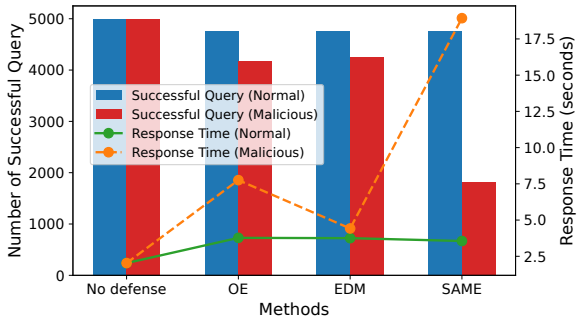

Figure 4: Comparison of flexibility as an add-on.

Model. For fair comparisons, the masked autoencoders of the three SAME versions share the same weight, respectively, on MNIST and CIFAR-10. In addition, we adopt the same training configuration as in the previous section unless otherwise specified. As shown in Figure 5, on MNIST, SAME and SAME−X achieve competitive performance. We guess this is due to the simplicity of the dataset, which leads to the near-perfect reconstruction performance of MAE. On CIFAR-10, SAME also shows leading performance, while the other two variants perform close to the same.

Effects of Malicious Ratio. We further consider the scenario where the malicious and normal queries are unbalanced. We increase the ratio of malicious queries from 0.01 to 0.9 gradually, as shown in Table 4. For both MNIST and CIFAR-10, SAME's AUROC remains stable as the proportion of malicious samples increases. The AUPR value increases along with the malicious ratio for both datasets. This trend underscores the model's improved ability to identify malicious queries as the prevalence increases in the dataset.

Effects of MAE Training Epochs. We evaluate the effect of the MAE training epochs on the reconstruction perfor-

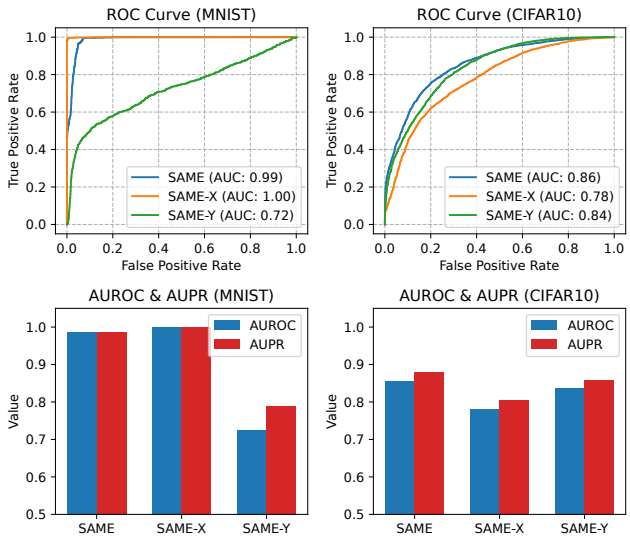

Figure 5: Detection performance of SAME and its variants on the MNIST and CIFAR-10 datasets.

Dataset	Ratio	AUROC↑	AUPR↑	FPR95↓	FPR90↓
MNIST	0.01	98.93	82.78	3.90	3.05
	0.05	99.07	95.42	3.82	3.12
	0.10	99.15	97.79	3.70	2.95
	0.30	99.23	99.30	3.65	2.80
	0.50	99.21	99.56	3.70	2.80
	0.90	99.21	99.71	3.70	2.80
CIFAR-10	0.01	86.91	50.91	61.20	46.62
	0.05	87.22	70.90	61.20	42.05
	0.10	87.64	79.53	60.45	40.58
	0.30	88.84	90.94	51.40	33.90
	0.50	89.39	94.23	47.20	31.45
	0.90	89.39	95.98	47.88	31.47

Table 4: The detection performance (%) of SAME under different proportions of malicious samples.

mance of clean and malicious samples, as shown in Figure 6. MAE was trained for 500 epochs on MNIST and CIFAR10 datasets, with a 50-epoch warm-up. Post 100 epochs, the MAE's reconstruction ability stabilized, showing a distinct average loss for different sample types. With increasing epochs, the reconstruction loss gap for Knockoff and DFME attacks widened, attributed to MAE's improving reconstruction of the original dataset but not the attacker dataset (OOD). For JBDA attacks, involving minor data perturbations, MAE improved in reconstructing both clean and malicious samples, underscoring the importance of Auxiliary Model deviation loss in the anomaly score function (Equation 6).

Effects of the MAE Embedding Size. For MAE, its embedding size represents the dimensionality of the condensed representation obtained by the encoder. This size affects the information transfer fidelity between the encoder and decoder, and thus the model's reconstruction accuracy. In the experiment, deviation loss from the Auxiliary Model is excluded to avoid interference. Results in Figure 7 show that increased embedding size improves SAME's detection performance.

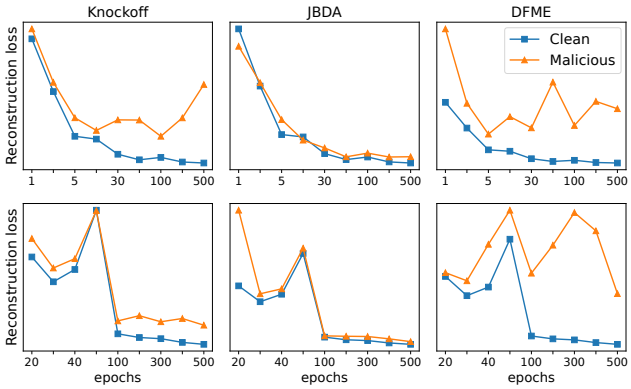

Figure 6: The reconstruction loss of the MAE for different categories of samples under different training epochs on MNIST (first row) and CIFAR-10 (second row) datasets.

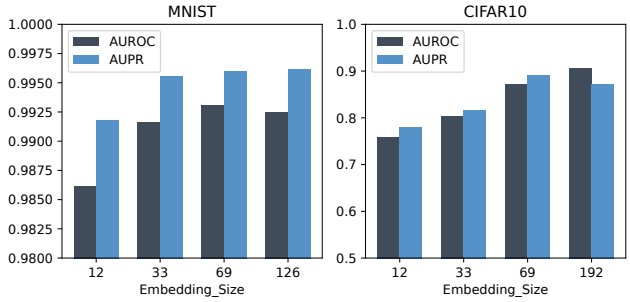

Figure 7: Detection performance of SAME under different MAE embedding sizes.

Conclusion

In this work, we propose a novel defense mechanism, SAME, to detect model extraction attacks. Compared to SOTA solutions, SAME does not require auxiliary datasets and demonstrates superior performance. Through comprehensive evaluations over common datasets, SAME displays high robustness against various extraction attacks under different query budgets. Moreover, our ablation studies confirm the effectiveness of each stage of our proposed solution, emphasizing the significance of the embedded representation in the Masked Autoencoder and its impact on detection accuracy. By integrating SAME with other active defenses, our end-to-end system exhibits improved defense capabilities. SAME enables maximum penalty for malicious queries while maintaining usability for normal users. In the future, we aim to explore other variants of SAME and further optimize it for specific deployment scenarios. Additionally, studying its applicability across other types of machine learning models will also be a valuable avenue for future research.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work is supported by the National Natural Science Foundation of China under Grant Nos. 61960206014 and 62121001, the National Research Foundation, Singapore, and the Cyber Security Agency under its National Cybersecurity R&D Programme (NCRP25-P04-TAICeN), and Singapore Ministry of Education (MOE) AcRF Tier 2 MOE-T2EP20121-0006. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of National Research Foundation, Singapore and Cyber Security Agency of Singapore.

References

Barbalau, A.; Cosma, A.; Ionescu, R. T.; and Popescu, M. 2020. Black-Box Ripper: Copying black-box models using generative evolutionary algorithms. *Advances in Neural Information Processing Systems*, 33: 20120–20129.

Chandrasekaran, V.; Chaudhuri, K.; Giacomelli, I.; Jha, S.; and Yan, S. 2020. Exploring connections between active learning and model extraction. In *29th USENIX Security Symposium (USENIX Security 20)*, 1309–1326.

Chen, K.; Guo, S.; Zhang, T.; Xie, X.; and Liu, Y. 2021. Stealing deep reinforcement learning models for fun and profit. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, 307–319.

Clanuwat, T.; Bober-Irizar, M.; Kitamoto, A.; Lamb, A.; Yamamoto, K.; and Ha, D. 2018. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718*.

Cohen, G.; Afshar, S.; Tapson, J.; and Van Schaik, A. 2017. EMNIST: Extending MNIST to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, 2921–2926. IEEE.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Dziedzic, A.; Kaleem, M. A.; Lu, Y. S.; and Papernot, N. 2021. Increasing the Cost of Model Extraction with Calibrated Proof of Work. In *International Conference on Learning Representations*.

He, K.; Chen, X.; Xie, S.; Li, Y.; Dollár, P.; and Girshick, R. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 16000–16009.

Hendrycks, D.; Mazeika, M.; and Dietterich, T. 2018. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*.

Jiang, W.; Li, H.; Xu, G.; Zhang, T.; and Lu, R. 2023. A Comprehensive Defense Framework Against Model Extraction Attacks. *IEEE Transactions on Dependable and Secure Computing*.

Juuti, M.; Szyller, S.; Marchal, S.; and Asokan, N. 2019. PRADA: protecting against DNN model stealing attacks. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, 512–527. IEEE.

Kariyappa, S.; Prakash, A.; and Qureshi, M. K. 2020. Protecting dnns from theft using an ensemble of diverse models. In *International Conference on Learning Representations*.

Kariyappa, S.; Prakash, A.; and Qureshi, M. K. 2021. Maze: Data-free model stealing attack using zeroth-order gradient estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13814–13823.

Kariyappa, S.; and Qureshi, M. K. 2020. Defending against model stealing attacks with adaptive misinformation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 770–778.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7): 3.

LeCun, Y.; Boser, B.; Denker, J. S.; Henderson, D.; Howard, R. E.; Hubbard, W.; and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4): 541–551.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Li, G.; Xu, G.; Guo, S.; Qiu, H.; Li, J.; and Zhang, T. 2022. Extracting Robust Models with Uncertain Examples. In *The Eleventh International Conference on Learning Representations*.

Li, J.; Chen, P.; He, Z.; Yu, S.; Liu, S.; and Jia, J. 2023. Rethinking Out-of-distribution (OOD) Detection: Masked Image Modeling is All You Need. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11578–11589.

Liu, B.; Ding, M.; Shaham, S.; Rahayu, W.; Farokhi, F.; and Lin, Z. 2021. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)*, 54(2): 1–36.

Lou, X.; Guo, S.; Li, J.; Wu, Y.; and Zhang, T. 2021. NASPY: Automated extraction of automated machine learning models. In *International Conference on Learning Representations*.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 427–436.

Orekondy, T.; Schiele, B.; and Fritz, M. 2019. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4954–4963.

Pal, S.; Gupta, Y.; Shukla, A.; Kanade, A.; Shevade, S.; and Ganapathy, V. 2020. Activethief: Model extraction using active learning and unannotated public data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 865–872.

Papernot, N.; McDaniel, P.; Goodfellow, I.; Jha, S.; Celik, Z. B.; and Swami, A. 2017. Practical black-box attacks

against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 506–519.

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.

Tramèr, F.; Zhang, F.; Juels, A.; Reiter, M. K.; and Ristenpart, T. 2016. Stealing machine learning models via prediction {APIs}. In *25th USENIX security symposium (USENIX Security 16)*, 601–618.

Truong, J.-B.; Maini, P.; Walls, R. J.; and Papernot, N. 2021. Data-free model extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 4771–4780.

Wang, W.; Yin, B.; Yao, T.; Zhang, L.; Fu, Y.; Ding, S.; Li, J.; Huang, F.; and Xue, X. 2021. Delving into data: Effectively substitute training for black-box attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4761–4770.

Yang, Y.; Hu, M.; Cao, Y.; Xia, J.; Huang, Y.; Liu, Y.; and Chen, M. 2023. Protect Federated Learning Against Backdoor Attacks via Data-Free Trigger Generation. *arXiv preprint arXiv:2308.11333*.

Yao, H.; Li, Z.; Weng, H.; Xue, F.; Ren, K.; and Qin, Z. 2023. FDInet: Protecting against DNN Model Extraction via Feature Distortion Index. *arXiv preprint arXiv:2306.11338*.

Yu, H.; Yang, K.; Zhang, T.; Tsai, Y.-Y.; Ho, T.-Y.; and Jin, Y. 2020. CloudLeak: Large-Scale Deep Learning Models Stealing Through Adversarial Examples. In *NDSS*.

Zhao, S.; Chen, K.; Hao, M.; Zhang, J.; Xu, G.; Li, H.; and Zhang, T. 2023. Extracting Cloud-based Model with Prior Knowledge. *arXiv preprint arXiv:2306.04192*.