Information Sciences 632 (2023) 269–284

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Explanation leaks: Explanation-guided model extraction attacks

Anli Yan [a,d], Teng Huang [b], Lishan Ke [b,*], Xiaozhang Liu [c,*], Qi Chen [b], Changyu Dong [b]

[a] *School of Cyberspace Security (School of Cryptology), Hainan University, China*
[b] *Institute of Artificial Intelligence and Blockchain, Guangzhou University, China*
[c] *School of Computer Science and Technology, Hainan University, China*
[d] *Pazhou Lab, Guangzhou, 510330, China*

ARTICLE INFO

Keywords:
Model extraction attacks
Explainable artificial intelligence
Black-box

ABSTRACT

Explainable artificial intelligence (XAI) is gradually becoming a key component of many artificial intelligence systems. However, such pursuit of transparency may bring potential privacy threats to the model confidentially, as the adversary may obtain more critical information about the model. In this paper, we systematically study how model decision explanations impact model extraction attacks, which aim at stealing the functionalities of a black-box model. Based on the threat models we formulated, an XAI-aware model extraction attack (XaMEA), a novel attack framework that exploits spatial knowledge from decision explanations is proposed. XaMEA is designed to be model-agnostic: it achieves considerable extraction fidelity on arbitrary machine learning (ML) models. Moreover, we proved that this attack is inexorable, even if the target model does not proactively provide model explanations. Various empirical results have also verified the effectiveness of XaMEA and disclosed privacy leakages caused by decision explanations. We hope this work would highlight the need for techniques that better trade off the transparency and privacy of ML models.

1. Introduction

Artificial intelligence (AI) has been ubiquitously and used in a wide range of tasks, ranging from image recognition, and natural language processing to object detection [1–3], etc. Recently, increasing demand for reliable AI systems addresses higher requirements for transparency and privacy, especially in scenarios where user and companies interest conflicts. On the one hand, users want their information protected and the prediction result more convincing. On the other hand, companies are reluctant in disclosing any details of their proprietary models, which, therefore, are usually encapsulated as a certain application program interface (API). As a result, many recent attempts emerged in improving the transparency of neural networks [4] and preserving training data as well as model confidentially [5]. To enhance reliability, explainable artificial intelligence (XAI) is employed to explain the logic behind every decision made by neural networks, as shown in Fig. 1. For AI systems, this is vital in improving performance, enhancing security, and accelerating deployment. Take image classification tasks, mainstreams for XAI are saliency maps [6], heat maps [7], and feature maps [8]. Since intuitive model explanations are provided with the prediction results, users can confirm the validity of model output more easily. However, despite attractive advantages, direct privacy leakage can be caused by sensitive knowledge carried out in model explanations [9–11]. In this paper, we focus on model extraction attacks (MEA), which aim to precisely copy model prediction patterns [12]. The crucial damage of MEA is beyond itself: it turns a black-box model into a white-box one, which significantly easies

[*] Corresponding authors.
E-mail addresses: kelishan@gzhu.edu.cn (L. Ke), lxzh@hainanu.edu.cn (X. Liu).

https://doi.org/10.1016/j.ins.2023.03.020
Received 17 June 2022; Received in revised form 22 February 2023; Accepted 2 March 2023

Available online 8 March 2023

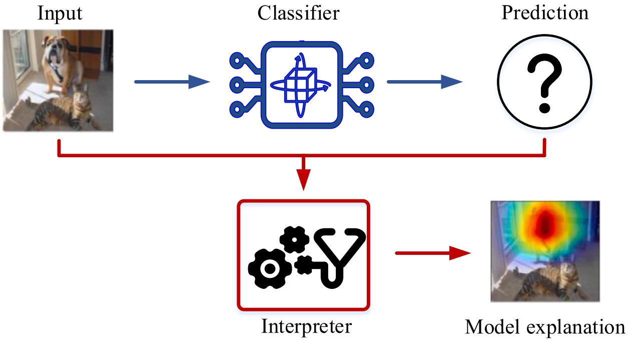


Fig. 1. Workflow of an explainable deep learning.

a series of downstream attacks, e.g., adversarial attacks [13], model inversion attacks [14], and membership inference attacks [15]. Nevertheless, how model decision explanations impact model extraction attacks remain rarely explored.

We first put forward a novel threat model, where model explanations are considered as auxiliary information to the adversary. We then design and propose XAI-aware Model Extraction Attacks (XaMEA) for such scenarios. Typically, model explanations contain decision boundaries, which very much simplifies model extraction attacks. Model predictions and explanations are properly fused in XaMEA, and thus can simultaneously be leveraged to train the encoder-decoder-based attack model. Furthermore, since the model explanation of image classification is the image of the same size as the query sample, the intuition of designing XaMEA is essentially to unearth the value information contained in the image. The attack in XaMEA can also be extended in more general cases since it is designed in a model-agnostic manner: the encoder-decoder structure flexibly adapts itself to various model architectures and corresponding explanation patterns. Besides, by applying black-box explanation techniques, models that do not proactively provide explanation information also fall into the scope of our attack. Our proposal is evaluated broadly, and various empirical results demonstrate its superiority. It outperforms Baseline and other state-of-art technologies [16,17]. Contributions of this paper are summarized as follows.

- We propose XaMEA, three XAI-aware model extraction attack architectures, the key is to fully unearth the model explanation knowledge by encoder-decoder structure. Their substitution models (i.e., copied from victim models) extracted through XaMEA have higher fidelity than traditional model extraction attacks. This emphasizes that the privacy risks of model explanations are significant.
- We further carry out XAI-aware model extraction attacks against non-explanation target models. It proves that the target model still faces the risk of the XAI-aware model extraction attack regardless of whether the target model shares explanations or not.
- We evaluate the attack effectiveness of XaMEA with an exhaustive set of experiments. Experimental results demonstrate that the attack accuracy of one of the proposed XAI-aware model extraction attacks is 12.75%-29.23% higher than the prediction-only model extraction attack method [16].

2. Background

2.1. Model extraction

Model extraction attacks destroy the confidentiality of ML models [18]. The typical attack procedure of model extraction attacks: an adversary first collects or synthesizes an initial unlabeled dataset. Then, the adversary queries the target model (also called the victim model) with inputs of their choice. Finally, the adversary leverages the attack dataset, which is annotated by the target model, to train a copy of the target model. The adversary's goal is to acquire a stolen replica that mimics the decision behavior of the target model.

Replicating a substitution model analogous to the structure and parameter of the target model is out of our scope. Because they are only suitable for shallow neural networks [19]. We also exclude model extraction attacks taking into account side-channel information [20]. In our elaborate designed threat model, the model not only supplies users with predictions but also provides auxiliary information, i.e., model explanations. The orientation of our efforts is how to make the best use of auxiliary information to perform model extraction attacks. Therefore, we explore the impact of explanations on model extraction attacks. Furthermore, our proposed XaMEA can also be extended to more general cases. Even if the target model does not furnish explanations, it is still the scope of our attack as described in Section 6.

2.2. Explainable AI

Explainable artificial intelligence (XAI) aims to lie in AI algorithms more transparent and reliable [21], especially in the medical, financial, and military fields. In line with the knowledge of the model, we taxonomize XAI technologies around two camps:

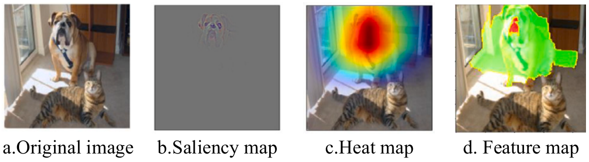


Fig. 2. Representation of explainable in the field of images.

model-related [22], i.e., knowing model gradients, structures, and parameters, etc, and model-agnostic [23], i.e., knowing the label or confidence score of the model output. One of the mainstream methods of model-related is to leverage the mapping of the convolution layer [24]. However, model-agnostic technologies speculate features that play an important role in model decisions in light of the change of predictions.

In this paper, we focus on image-based XAI privacy security. There are three representations of explanations in the domain of image as shown in Fig. 2. Image-based XAI can be transformed into arbitrary maps. Therefore, the tailored intuition of XaMEA is to make full use of triples (X, E_t, Y_t). X is the original image of the attack dataset. E_t and Y_t correspond to the explanations and labels of the attack dataset, respectively, which are gained by the target model.

3. Our method

3.1. Overview

XaMEA is a model extraction attack technology, which aims to explore the privacy risks of explanation leakage. It operates by (1) collecting the attack dataset, (2) querying explanations and labels via the target model, and (3) training the substitution model to predict. XaMEA's framework is composed of the following components.

- **Attack Dataset Collection (ADC):** It collects attack samples as capital for an adversary to launch the model extraction attack.
- **Target Model Query (TMQ):** It queries the attack dataset to the target model. Since then, the attack dataset is annotated with labels and explanations via the target model.
- **Substitution Model Train (SMT):** It leverages the annotated attack dataset to train the substitution model with decisions analogous to the target model.

We note that the ADC and TMQ components are general, so they can be reapplied as a general XAI-aware model extraction attack. The core component of XaMEA is SMT, for which we design three different schemes to better excavate the knowledge of explanations. Note that the schemes we designed are based on transfer learning. The first scheme is called the double encoder transfer model extraction attack (DET). The second scheme is called the logits and encoder transfer model extraction attack (LET). The last scheme is called the single encoder transfer model extraction attack (SET). The first and third schemes focus on utilizing spatial features between images. The second scheme considers both the spatial features of images and the entanglement features between images and labels. Before introducing the detail of the three pipelines, we first understand the setup of the threat model.

Threat Models. Consider a well-trained target model that is deployed on a cloud platform and provides an application program interface (API) for queries. To better its service, the company behind this model returns predictions and coupled explanations for users, e.g., saliency maps. The adversary in the public environment aims at stealing the functionalities of the target model. He queries the target model by his local dataset (also called the attack dataset) and observes the corresponding output pattern. In the scenario above, the output pattern for each queried image contains a prediction vector and explanation tensor. The explanation tensor usually has a consistent size to the input image.

Adversary Knowledge. The adversary has black-box access to the target model. The adversary obtains label-only and coupled explanations through the target model query. The adversary has the same distribution of dataset as the target model training set. Moreover, the adversary trying to reconstruct the substitution model does not know the model type of the target model. In the following, unless otherwise specified, the adversary's knowledge is as described above.

3.2. Double encoder transfer model extraction attack (DET)

The substitution model obtains by the double encoder transfer model extraction method. It is composed of two encoders and a classifier, as shown in the red box in Fig. 3. One encoder (M_a^e) is derived from an autoencoder (M_a) trained by query images (X), and its reconstruction loss is $L_a = (M_a(X) - X)^2$. The other encoder (M_i^e) is derived from an autoencoder (M_i) trained with the query images, and its reconstruction loss is $L_i = (M_i(X) - E_t)^2$. E_t is the explanations about the query images provided by the target model. Both L_a and L_i are mean-square error (MSE).

When the two autoencoders have been trained, the adversary transfers their encoders to the first half of the substitution model. The second half of the substitution model is a classifier. The input to the classifier is a concatenation of the output of two encoders $(M_a^e(X) \odot M_i^e(X))$. The classifier can be MLP or CNN, etc. The substitution model (M_s) with cross-entropy loss $L_s(M_s(X), Y_t)$, where Y_t is the predicted label from the target model. The training of the substitution model adjusts the parameters of the classifier according to the loss function L_s, and the two encoders transferred also carry out fine-tuning.

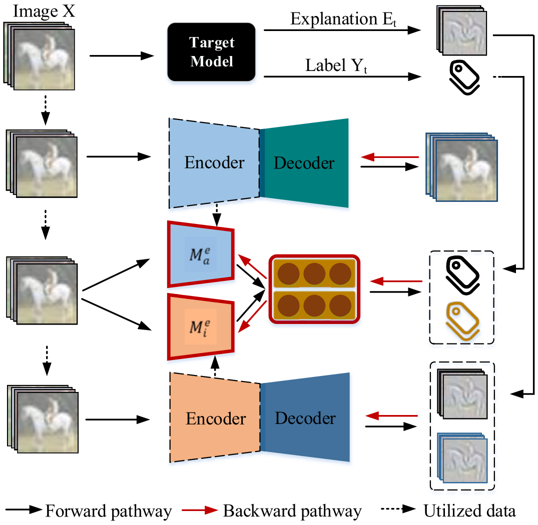


Fig. 3. A schematic view of DET pipeline.

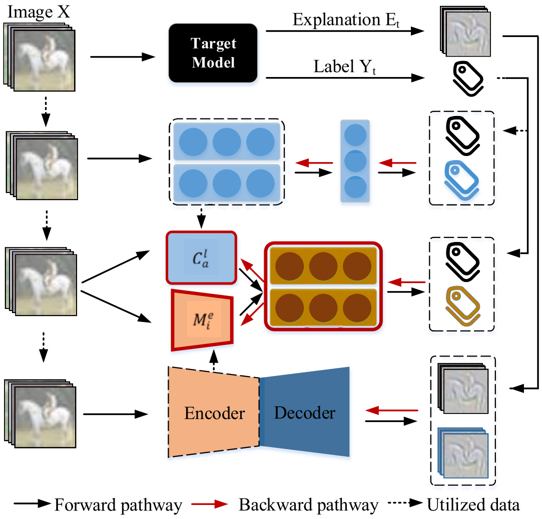


Fig. 4. A schematic view of LET pipeline.

3.3. Logits and encoder transfer model extraction attack (LET)

The logits and encoder transfer model extraction attack not only transfer the network structure and parameters before the softmax layer of a classifier, but also transfer the encoder of an autoencoder. The framework is shown in Fig. 4, and the red box constitutes the substitution model. As depicted in Fig. 4, first an adversary trains the front half part of the substitution model. One is a classifier (C_a) trained using query images (X) and the prediction label (Y_t) provided by the target model. The C_a is trained with cross-entropy loss, $L_a(C_a(X), Y_t)$. The other is an autoencoder (M_i) trained using the query images, its reconstruction loss, i.e., MSE, is $L_i = (M_i(X) - E_t)^2$. E_t is the explanations feedback from the target model.

After the adversary has trained the C_a and M_i, the C_a transfers the network structure and parameters before the neural network softmax layer (C_a^l), the M_i transfers its encoder (M_i^e). Subsequently, the adversary supplements the latter part of the substitution model structure, and trains the overall substitution model. The latter part of the substitution model is essentially a classifier, whose input is the concatenation of $(C_a^l(X) \odot M_i^e(X))$. The training loss of the substitution model (C_s) is cross-entropy $L_s(C_s(X), Y_t)$. In the overall training process of the substitution model, the parameters of the transferred module can be adjusted according to the loss of L_s.

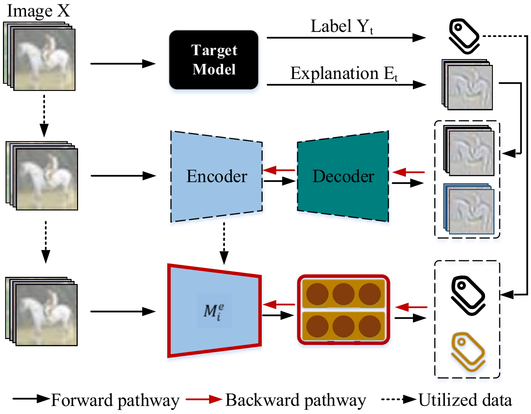


Fig. 5. A schematic view of SET pipeline.

Table 1
Accuracy of target model.

Target Models	Datasets	Accuracy
VGG19	MNIST	98.70%
VGG19	Fashion-MNIST	91.45%
ResNet50	CIFAR-10	92.03%
DensNet161	CIFAR-100	72.02%

3.4. *Single encoder transfer model extraction attack (SET)*

In the last type of attack scenario, an adversary also uses an autoencoder and transfers its encoder to be element of the substitution model. Compared with the DET in Section 3.2, this method only needs to transfer one encoder. Hence, our last method is named single encoder transfer model extraction attack method (SET). To build the substitution model, the adversary first trains the autoencoder (M_i), its input is query images (X). The loss function of M_i is mean-square error, $L_i(M_i(X), E_t)$. E_t is the explanations corresponding to the query images provided by the target model. After the M_i is trained in the first stage, the adversary transfers the encoder (M_i^e) of the M_i to the substitution model as a sub-module. Another sub-module of the substitution model (M_s) is a classifier whose training loss function is cross-entropy $L_s(M_s(X), Y_t)$, where Y_t is the prediction label provided by the target model to the adversary's query images. The framework diagram of SET is shown in Fig. 5, in which the red boxes are the components of the M_s.

4. Experimental design and implementation

4.1. *Datasets description*

We evaluate the experiment using four publicly available image datasets, MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100. MNIST and Fashion-MNIST are the grayscale images of 28 × 28 with 10 labels, where the example number of the training set is 60,000, and the example number of the test set is 10,000. CIFAR-10 and CIFAR-100 are three-channel color images of 32 × 32, where the example number of the training set is 50,000 and the example number of the test set is 10,000. CIFAR-10 is the dataset with 10 labels, while CIFAR-100 is the dataset with 100 labels. Note that we sometimes abbreviate Fashion-MNIST to FMNIST for the convenience of description.

In our experiment, all training set examples are used as the training data of the target model. The test set is owned by the adversary. The adversary uses 80% of the test set as the attack dataset, i.e., 8,000 samples, and 20% of the test set, i.e., 2,000 samples, is used to test the label similarity between the substitution model and the target model (see Table 1 for the accuracy of target model).

4.2. *Target models*

There are three types of target models that we draw up. For MNIST and Fashion-MNIST datasets, the target model is trained by the visual geometry group network (VGG19). For the CIFAR-10 dataset, the target model is the residual network (ResNet50). For the CIFAR-100 dataset, the target model adopts the densely connected convolutional network (DensNet161). The task accuracy of the target model is shown in Table 1.

4.3. Explanation types

We assume that the target model is two camps, model-related and model-agnostic. For model-related XAI techniques, we select the gradient-based method (GRAD) [22] and gradient-weighted class activation mapping method (Grad-CAM) [24] as representatives. In the other camp, we select perturbation strategy (MASK) [25] and local model explanations generated by LIME [26] as representatives.

4.4. Evaluation metrics

We assume that the target model is M_t and the substitution model is M_s. Our goal is to minimize $S(M_t, M_s)$, where $S(\cdot)$ is the similarity function. In this paper, we consider only label agreement $S(M_t(X), M_s(X)) = (argmax(M_t(X)) = argmax(M_s(X)))$, where X is test set. In other words, the label agreement is also called attack accuracy. $S(M_t(X), M_s(X)) = accuracy(Y_t, Y_t')$, where Y_t is the set of predicted labels of M_t for X, Y_t' is the set of predicted labels of M_s for X.

4.5. Experiment setup

To evaluate the performance of XaMEA, i.e., DET, LET, and SET, we design seven sets of experiments: parameter analysis, the effectiveness of XAI-aware attacks, attacking target models with different XAI types, attacking target models with different complex tasks, attacking target models with different datasets, attacking target models with different budgets, and evaluation of XAI-aware attacks against defense target models.

Parameter analysis. The major purpose is to evaluate the attack efficiency of XaMEA. The three XaMEA attack pipelines we proposed have one thing in common: some components of their substitution model are transferred from other models. As described in Section 3. To construct a substitution model, XaMEA first requires training the transferred model and then transferring some elements of the transferred model to the substitution model. Subsequently, we design a convolution network structure, which constitutes the other parts of the substitution model. Finally, we train the substitution model to adjust the overall internal parameters.

Effectiveness of XAI-aware attacks. We evaluate XaMEA by comparing it with a Baseline and two state-of-the-art model extraction attack technologies proposed in [16,17]. The Baseline utilizes attack datasets to train a substitution model directly. PRADA [16] uses Jacobian-based dataset augmentation to perform model extraction attack. It is worth mentioning that the Baseline and the state-of-the-art model extraction attack are label-only and confidence scores respectively. Besides, we also capture an empirical comparison with the approach of Milli et al. [17]. This work is closely related and addresses the same objective as this paper.

Attacking target models with different XAI types. We aim to measure the degree of privacy risks of different explanation types. The explanation types we choose are GRAD, Grad-CAM, MASK, and LIME as mentioned in Section 4.3.

Attacking target models with different complex tasks. The datasets we select to evaluate included MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100. The order of task complexity between them is MNIST < Fashion-MNIST < CIFAR-10 < CIFAR-100. We leverage to observe the difference in XaMEA attack performance of the target model under different task complexity.

Attacking target models with different datasets. Our goal is to assess the attack performance of XaMEA under different knowledge of the target model training datasets. In this subsection, we assume that there are three types of adversaries: superpower, power, and weak. Superpower means that the adversary can obtain the same data sample as the target model training set. Power means that the adversary can steal a subset of the target model training set. Weak means that the adversary can only collect the attack dataset with the same distribution as the target model training set. It is worth noting that the adversary is the weak type in other experimental evaluation settings.

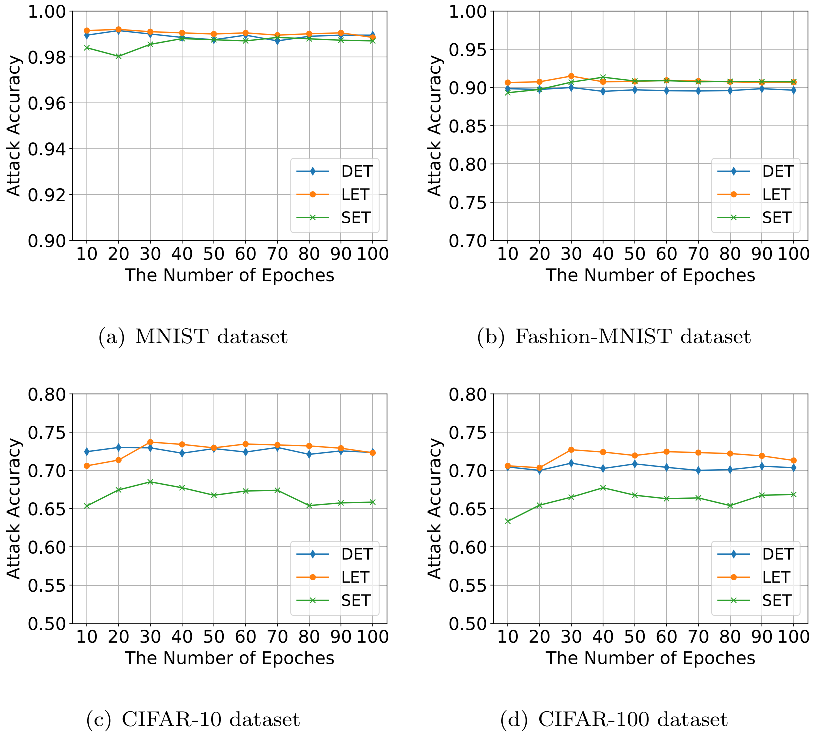
Attacking target models with different budgets. We presume that there are limitations like a fixed number of queries per day or a fixed cost per query, requiring an adversary to perform as few queries as possible. Since each query gives additional information (in the form of an explanation), we expect that the proposed XAI-aware attacks require much fewer queries than the label-only Baseline.

Evaluation of XAI-aware attacks against defense target models. In practice, explanations usually provide a processed version of the saliency map, instead of raw saliency maps. We also test the explanations processed with adversarial example technologies, i.e., FGSM [27] and AutoPGD [28].

5. Experimental results

5.1. Parameter analysis

Intuitively, the substitution model constructed by XaMEA needs expensive training costs. Because the XaMEA requires training multiple neural networks to construct the substitution model. Surprisingly, the proposed XaMEA attacks do not require much effort from an adversary. The reason is that the adversary training transferred model does not require too many epochs. As shown in Fig. 6, the abscissa represents the number of epochs of training the transferred model, and the ordinate represents the attack accuracy of the substitution model. With the increase of epochs, the changes of different measurement curves are basically consistent, i.e., as epochs increase, the attack accuracy of the three XaMEA attack methods is almost unchanged. We prove through experiments that the transferred model can maintain high performance without a large number of training epochs.



(a) MNIST dataset

(b) Fashion-MNIST dataset

(c) CIFAR-10 dataset

(d) CIFAR-100 dataset

Fig. 6. Parameter setting according to attack accuracy.

5.2. *Effectiveness of XAI-aware attacks*

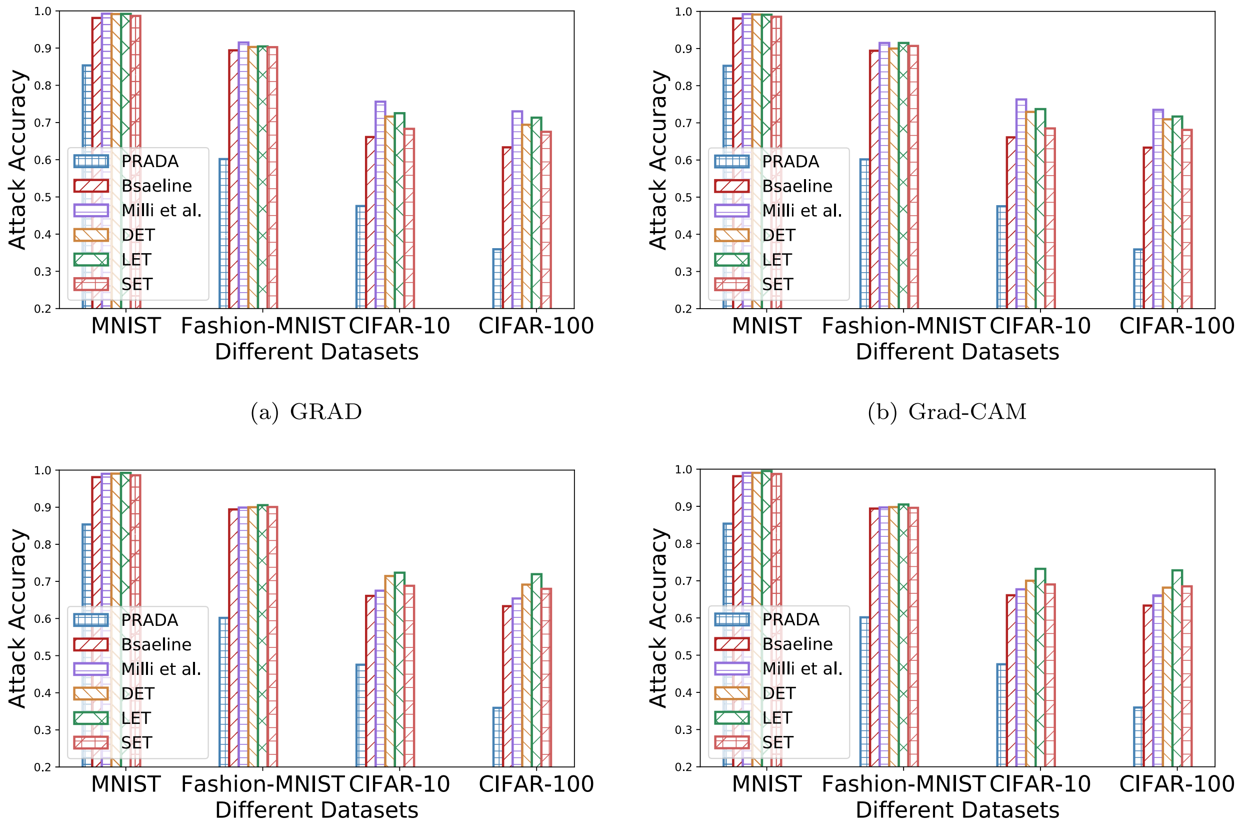
For different model explanation methods, the performance trend of XaMEA attacks is basically consistent (see Fig. 7). The attack performance of attack methods is improved in the order: Baseline < SET < DET < LET. For MNIST and Fashion-MNIST datasets, the attack accuracy of LET is 0.85%-1.4% and 1.1%-2.32% higher than the Baseline. What is more impressive is that for CIFAR-10 and CIFAR-100 datasets, the attack accuracy of LET is improved by 6.25%-13.6%, and 7.97%-9.47%, respectively, compared with the Baseline. DET's attack accuracy is 0.91%-1.05%, 0.42%-0.9%, 3.9%-6.85%, and 5.81%-7.62% higher than the Baseline in each of the four datasets. Moreover, the attack accuracy of SET is 0.45%-0.62%, 0.24%-1.31%, 2.23%-3.01%, and 4.17%-5.17% higher than the Baseline in the four datasets.

On the other hand, we find that the attack performance of PRADA is the lowest, even weaker than the Baseline we designed. We analyze the reasons for this phenomenon: PRADA utilizes the Jacobian matrix to synthesize sample training substitution model, which requires expensive query and the attack dataset with large initial cardinality. To ensure the fairness of the experiment, PRADA still limits the number of queries to 8,000, and the number of the initial attack dataset is also 8,000. The Baseline and the state-of-the-art model extraction attack technology perform poor than XaMEA because they capture the least attack information.

Furthermore, we separately compare the work of Milli et al. [17] with XaMEA. The work of Milli et al. has the same goals as XaMEA, i.e., using explanations to perform model extraction attacks. For GRAD and Grad-CAM explanation techniques, the attack accuracy of Milli et al. is higher than the optimal LET in XaMEA. However, for MASK and LIME explanation techniques, the attack accuracy of Milli et al. is very poor. The reason for this phenomenon is that MASK and LIME are explanation techniques based on perturbation strategy, so the explanation is not unique. Note that the work of Milli et al. can only achieve the desired attack effect when the adversary knows the explanation technique used by the target model. It is also demonstrated that the model explanation can make the model extract attack have higher fidelity, thus verifying that explanations leak privacy.

5.3. *Attacking target models with different XAI types*

The purpose of this subsection is to assess the degree of privacy risk of four different XAI types. The experimental statistics are shown in Fig. 8. Regardless of XaMEA attack methods, MNIST and Fashion-MNIST datasets have a relatively small gap in the attack accuracy of the four different model explanation technologies. We focus on the analysis of the CIFAR-10 dataset as an example. We first analyze the impact of DET on the four model explanation technologies, and Grad-CAM has the highest attack accuracy. Subsequently, we turn our attention to the performance of LET on different XAI types, Grad-CAM's attack accuracy is always optimal. For the SET attack method, its most prominent attack accuracy is LIME. Although LIME has the highest attack accuracy, the difference between Grad-CAM and LIME is tiny. Similarly, for the work of Milli et al. [17], the same conclusion is reached, and Grad-CAM has the greatest attack accuracy. In particular, it is emphasized that the work of Milli et al. has certain limitations and requires the



(a) GRAD

(b) Grad-CAM

(c) MASK

(d) LIME

Fig. 7. Comparison of attack accuracy between XaMEA and other prediction-only technologies.

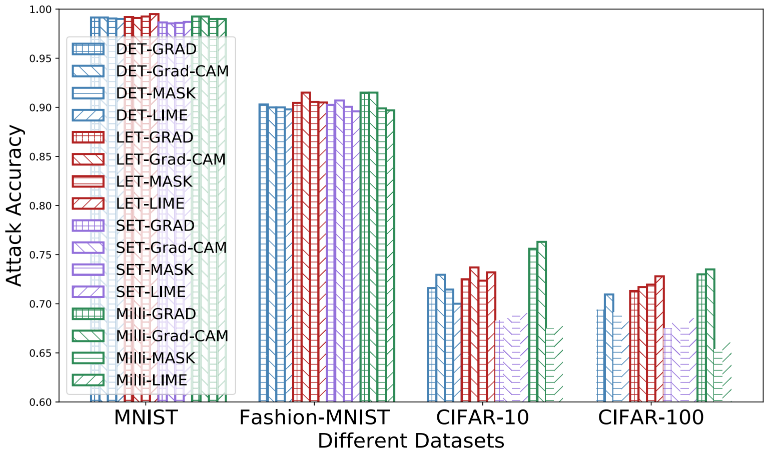


Fig. 8. Comparison of attack accuracy of different XAI types.

adversary to know more knowledge, i.e., the explanation technology adopted by the target model. Through the above analysis, we conclude that Grad-CAM is the winner, which means Grad-CAM has the greatest risk of privacy leakage.

5.4. Attacking target models with different complex tasks

We reanalyze the attack performance of XaMEA from the perspective of target models' task complexity (see Fig. 7). According to the complexity of tasks, we can conclude that different attack methods have similar statistical results, i.e., the attack performance of simple tasks is much higher than that of complex tasks. Under the GRAD explanation technology, LET obtains 99.2%, 90.45%,

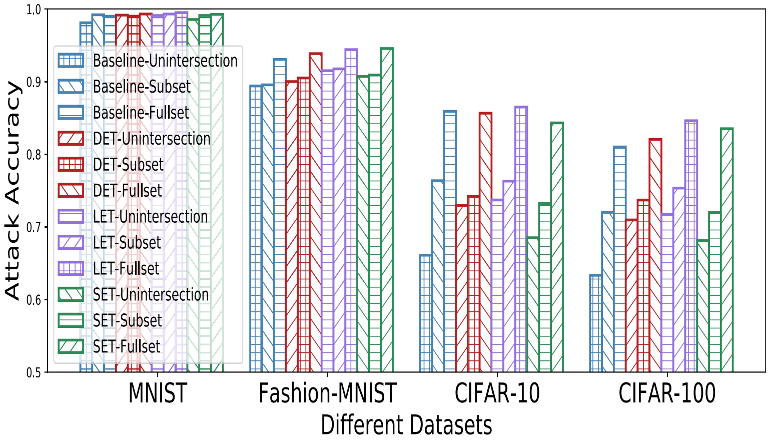


Fig. 9. Comparison of attack accuracy of XaMEA attack methods against Grad-CAM explanation technology under different attack datasets.

72.5%, and 71.3% attack accuracy respectively in MNIST, Fashion-MNIST, CIFAR-10 and CIFAR-100 datasets. Moreover, the more complex the task, the more obvious the improvement effect of XaMEA. We compare the optimal LET in XaMEA with Grad-CAM. Statistically, we can find that their attack performance gap on the four datasets (i.e., from simple to complex) is 1.0%, 2.1%, 7.6%, and 8.37%. From the analysis of the above results, we can conclude that when the target model deals with more complex tasks, our proposed XaMEA has greater superiority, especially LET.

5.5. Attacking target models with different datasets

We evaluate whether the more the intersection of the attack dataset and the target model training dataset can improve the model extraction performance. As expected, the more knowledge the adversary has, the greater the privacy risk. The performance of attack methods is improved successively: Unintersection < Subset < Fullset (see Fig. 9). Subset means that the attack dataset is a subset of the target model training set. Fullset means that the attack dataset is the same as the target model training set. Unintersection means that the attack dataset does not intersect with the target model training set, and is only in the same distribution as the target model training set.

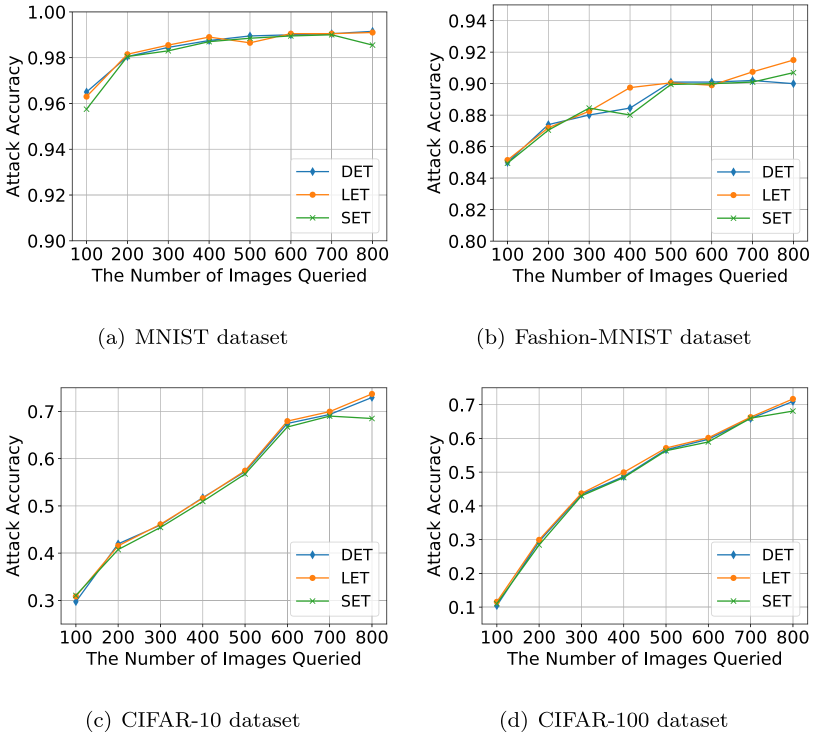
One interesting finding is that Baseline-Subset is slightly better than XaMEA attack methods (DET-Unintersection, LET-Unintersection and SET-Unintersection) under certain conditions. For example, the performance difference between LET-Unintersection and Baseline-Subset on MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100 datasets is 0.01%, 1.95%, 2.65%, and 0.3% respectively. Therefore, XaMEA can make up for the decline in attack accuracy caused by the lack of adversary knowledge. The other finding is quite remarkable given that XaMEA attack methods have better universality. For the CIFAR-10 dataset, the gap between Baseline-Unintersection and Baseline-Subset is 10.25%. However, for XaMEA attack methods, the gap between Unintersection and Subset is 3.45%-4.87%. We speculate that the reason for this phenomenon is that XaMEA attack methods are less sensitive to whether the attack dataset is the training samples of the target model.

5.6. Attacking target models with different budgets

We evaluate the attack accuracy of XaMEA versus Baseline under different query budgets (see Fig. 10). With the increase of query images, no matter what kind of attack method, its attack accuracy is constantly improving, especially for CIFAR-10 and CIFAR-100 datasets. Note that the image we queried is randomly sampled from the attack dataset and does not employ any outstanding sample selection algorithms. For the CIFAR-10 dataset, when the attack accuracy of the Baseline is 66.1%, the adversary is required to query 8,000 images from the target model. However, for XaMEA attack methods, to achieve the attack accuracy around of 66%, DET, LET and SET require the adversary to query 6,000 images from the target model. In addition, when the attack accuracy of the adversary reaches 63.33%, the difference between the query images of the Baseline and XaMEA is 1,000, which is under the CIFAR-100 dataset. Through the experimental analysis of Fig. 10, as we speculate, XaMEA obtains richer knowledge from explanations, so it saves the adversary's budgets.

5.7. Evaluation of XAI-aware attacks against defense target models

As reported in Table 2, the explanation preprocessed by the defender has no effect on XaMEA. For different datasets, the attack accuracy of XaMEA fluctuates slightly under clean explanations and perturbed explanations. For example, under the CIFAR-10 dataset, the difference in attack performance of LET against the defense and non-defense target models is 0.35%-0.45%. Under the CIFAR-100 dataset, the attack performance gap between the defense and non-defense target models of LET is 0.05%-0.1%. For other XAI techniques (i.e., GRAD, MASK, and LIME), their behavior is analogous to Grad-CAM. Therefore, a simple defense cannot prevent XaMEA attacks. We need to explore new technology to mitigate privacy leakage caused by explanations.



(a) MNIST dataset

(b) Fashion-MNIST dataset

(c) CIFAR-10 dataset

(d) CIFAR-100 dataset

Fig. 10. Impact of query budget on attack accuracy. Grad-CAM as an example.

Table 2
Comparison of attack accuracy between defense and non-defense target models. Grad-CAM as an example.

Dataset	Method	DET	DET_d	LET	LET_d	SET	SET_d
MNIST	FGSM	99.15%	99%	99.1%	99.15%	98.55%	98.44%
	AutoPGD		99.1%		99.05%		98.65%
FMNIST	FGSM	90%	89.55%	91.5%	91.45%	90.7%	90.6%
	AutoPGD		89.3%		90.95%		90.45%
CIFAR-10	FGSM	72.95%	72.2%	73.7%	74.05%	68.5%	68.47%
	AutoPGD		72.3%		73.25%		68.5%
CIFAR-100	FGSM	70.95%	69.95%	71.7%	71.65%	68.1%	68.05%
	AutoPGD		70.45%		71.8%		68.01%

Discussion: We compared the attack accuracy of XaMEA and other state-of-the-art model extraction attack technologies [16,17] in different dimensions. We come to the following conclusion.

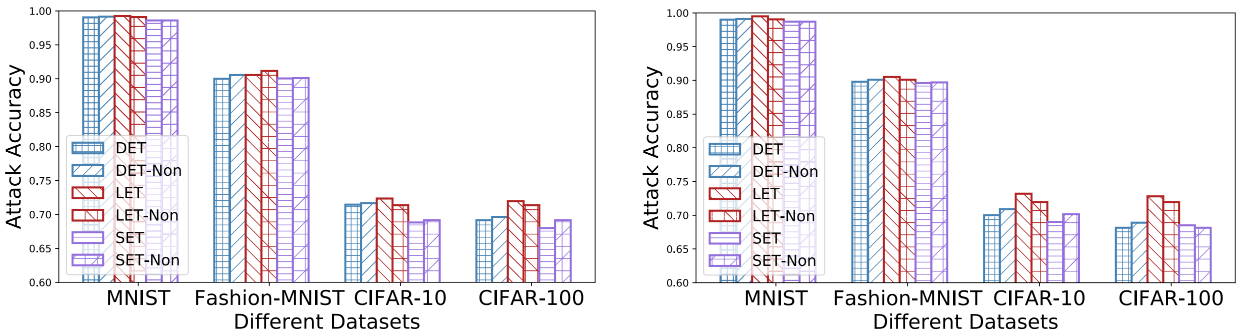
(1) Even though XaMEA requires the construction of multiple transferred models, it does not affect the attack efficiency of XaMEA (see Section 5.1). Because the transferred model does not require too many epochs, which can still keep the attack accuracy of the substitution model at a high level (see Fig. 6).

(2) XaMEA performs better than popular model extraction attack technologies (see Section 5.2). In particular, the LET method of XaMEA has 3.75%-4.15%, 30.28%-31.33%, 24.8%-25.65%, and 35.37%-36.87% higher attack accuracy than PRADA [16] on four datasets (i.e., MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100). Furthermore, compared with the work of Milli et al. [17], it is slightly better than LET only if the adversary knows the XAI technique employed by the target model and the XAI technique is stable for each image. Otherwise, the proposed schemes by us are superior to Milli et al. (see Fig. 7).

(3) The technology with better explanations faces more privacy risks (see Section 5.3). By analyzing different XAI technologies, we conclude that Grad-CAM [24] may reveal more privacy (see Fig. 8).

(4) The more complex the task of the target model, the more prominent the superiority of XaMEA (see Section 5.4). Specifically, for the MNIST dataset, XaMEA only improves the attack accuracy by 0.5%-1.4% on the basis of the Baseline, while for the CIFAR-100 dataset, XaMEA improves by 4.17%-9.47% (see Fig. 7).

(5) XaMEA is insensitive to the knowledge of target model training dataset, compared with other model extraction attack technologies, i.e., the Baseline (see Section 5.5). The difference in attack accuracy between Baseline-Unintersection and



(a) MASK

(b) LIME

Fig. 11. Comparison of attack accuracy of XaMEA attack methods under model-agnostic XAI technologies.

Baseline-Subset is 0.15%-10.25%. However, the gap in attack accuracy between LET-Unintersection and LET-Subset is only 0.25%-3.64% (see Fig. 9).

(6) XaMEA requires far fewer queries than the Baseline, under the condition of achieving the analogous attack accuracy (see Section 5.6). Because explanations contain much more information than the predicted labels. XaMEA has 2,000 fewer queries than the Baseline when the attack accuracy reaches about 66% on the CIFAR-10 dataset (see Fig. 10).

(7) Even if the target model is defended before releasing explanations, the attack performance of XaMEA is still not affected (see Section 5.7). Whether the explanations of the target model are perturbed by FGSM or AutoPGD, there is little change in the attack accuracy of XaMEA (see Table 2).

6. Non-explanation target models

We consider that not all target models provide explanations to users. To extend XaMEA to general scenarios, the non-explanation target model also falls into the scope of our attack. We assume that the adversary obtains the explanations by leveraging model-agnostic XAI technologies. In this scenario, the adversary also has black-box access to the target model. The adversary acquires confidence scores through the target model query. We evaluate the attack accuracy and make statistics on the query of target models. Furthermore, we set the target model into two types, defense and non-defense. For the defense target model, the confidence score provided to the user is perturbed without affecting the prediction. However, the other directly offers the clean confidence score. Note that we leave aside the target model of stopping the service due to abnormal user behavior.

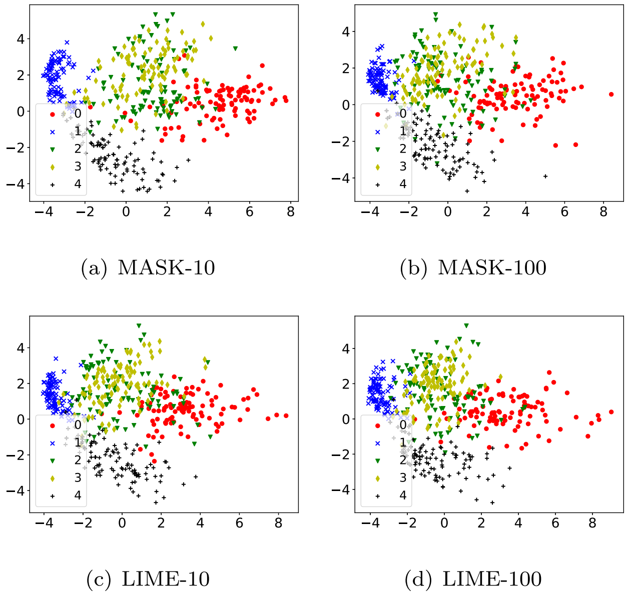
6.1. Effectiveness of XAI-aware attack on non-explanation target models

We apply two model-agnostic XAI techniques, MASK and LIME, to evaluate. As reported in Fig. 11, we compare the attack performance of XaMEA attack methods using the same model-agnostic XAI techniques between adversary and target models. In Fig. 11, the Non-suffix is the attack performed by the adversary against the non-explanation target model. On the contrary, it is the attack on the explanation target model. Both MASK and LIME techniques require multiple queries to obtain the explanation for a given sample. Because they observe the changes predicted by the target model via perturbing the image, they derive the important features of the model decision. Considering that the adversary's goal is to achieve a model extraction attack with the least budget. Therefore, for the non-explanation target model, the adversary only performs 10 queries on the target model to obtain the explanation of the specified sample. Since the total number of samples in the attack dataset for training the substitution model is 8,000, the total number of queries made by the adversary is 80,000. However, for the explanation target model, it carries out 100 queries for each sample to provide the explanation.

Experiments on multiple datasets show that XaMEA can be extended to general attack scenarios, i.e., even for non-explanation target models. Especially for the LET method, the gaps between LET and LET-Non on the CIFAR-10 dataset are 1.0% (see Fig. 11(a)) and 1.25% (see Fig. 11(b)), respectively. The difference in attack accuracy between DET and DET-Non on the Fashion-MNIST dataset is 0.55% (see Fig 11(a)). The attack accuracy gap between SET and SET-Non on the MNIST dataset is 0.03% (see Fig. 11(b)). Through the comprehensive analysis of these experiments, we can conclude that regardless of whether the target model provides explanations, there is still a risk of more privacy leakage due to model-agnostic explanation technologies.

6.2. Query efficiency analysis of XAI-aware attacks

We test the query efficiency of XaMEA using the MASK technology, and the results are given in Table 3. XaMEA attack methods have achieved impressive performance over the different number of individual sample queries. For the MNIST dataset, the largest attack accuracy gap of XaMEA is 0.9%. For the Fashion-MNIST dataset, the maximum attack accuracy differences of DET, LET, and SET are 1.0%, 1.55%, and 0.5%, respectively. Similarly, we analyze the attack performance of XaMEA on the CIFAR-10 dataset, 0.95%, 2.35%, and 1.63% are the largest gaps among DET, LET, and SET. It also has a question that deserved serious consideration,



(a) MASK-10 (b) MASK-100

(c) LIME-10 (d) LIME-100

Fig. 12. The explanation data points, which are obtained by different individual sample query times, are projected into a 2D space using principal component analysis (PCA). The MNIST dataset is an example.

Table 3
The impact of the number of queries on the attack accuracy (%) of XaMEA leverages MASK technology.

Dataset	Method	The number of individual sample queries									
		10	20	30	40	50	60	70	80	90	100
MNIST	DET	99.15	98.85	99.05	98.90	99.15	99.00	98.85	98.85	98.80	99.05
	LET	99.10	99.20	99.15	99.40	99.20	99.15	99.25	99.35	99.15	99.25
	SET	98.70	98.50	98.60	98.80	98.50	98.70	98.60	98.55	98.85	98.60
FMNIST	DET	90.55	89.55	90.20	90.15	90.30	90.50	90.45	90.35	90.25	90.00
	LET	91.15	91.30	91.95	90.70	91.45	91.45	90.55	90.55	90.40	90.55
	SET	90.10	89.70	90.15	90.00	89.85	89.55	90.30	89.90	90.10	90.05
CIFAR-10	DET	71.65	71.85	71.65	71.10	71.05	71.50	71.10	70.90	71.40	71.45
	LET	71.35	72.40	72.55	72.65	73.55	73.70	73.40	73.55	73.25	72.35
	SET	69.15	69.75	69.45	70.01	70.45	70.15	69.95	70.25	70.15	68.82
CIFAR-100	DET	69.65	68.85	70.65	69.10	69.05	70.05	69.05	70.90	69.40	69.14
	LET	71.35	71.40	71.45	71.65	71.80	71.95	71.75	71.80	71.25	71.95
	SET	69.15	68.45	68.65	68.01	68.45	68.15	69.05	68.25	68.15	68.00

i.e., attacks with fewer queries compare well to attacks with more queries. In particular, in the LET method under the CIFAR-10 dataset, the reduced number of queries does not significantly degrade the quality of the substitution model, but sometimes improves it.

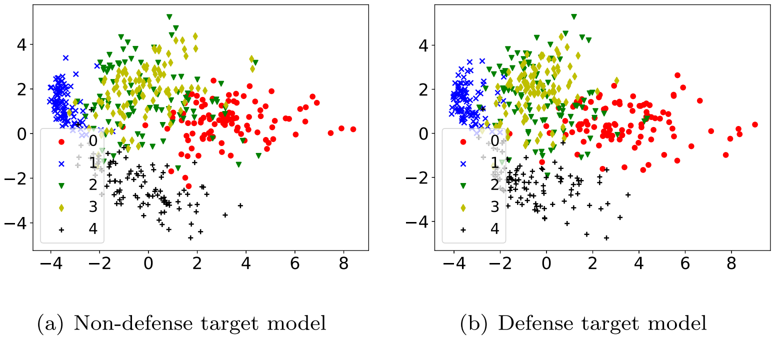
To this end, we leverage principal component analysis (PCA) to map the explanation obtained under different individual sample queries to a 2D space. For the MASK technology, even under different individual sample query times, the distribution of explanations is consistent (see Fig. 12(a) and Fig. 12(b)). Similarly, the LIME technology is the same statistical result (see Fig. 12(c) and Fig. 12(d)). In Fig. 12, the suffix 10 indicates that the number of individual sample queries is 10, otherwise, it is 100. This further sheds light on the severe privacy risks caused by XAI-aware model extraction attacks against ML models, i.e., excellent attack accuracy can be achieved without expensive query budgets.

6.3. Evaluation of XAI-aware attacks against defense non-explanation target models

In our experimental evaluation, we focus on evaluating XaMEA attacks against the defense target model performed using the LIME technology. The posterior probability perturbation strategy we designed is to add noise to the posterior without changing the final prediction [29]. The statistical results are shown in Table 4. We find that no matter what XaMEA attack methods or datasets, the XaMEA we proposed has excellent attack accuracy for both defense and non-defense target models. We analyze the causes of this phenomenon. Therefore, we map the explanation obtained from the defense and non-defense target models to 2D space, using

Table 4
Comparison of attack accuracy between defense and non-defense target models for non-explanation target models.

Dataset	Method	Non-defense	Defense
MNIST	DET	99.05%	98.95%
	LET	99.25%	99.20%
	SET	98.60%	98.70%
FMNIST	DET	90.00%	90.15%
	LET	90.55%	91.05%
	SET	90.05%	89.65%
CIFAR-10	DET	71.45%	71.60%
	LET	72.35%	70.65%
	SET	68.82%	68.75%
CIFAR-100	DET	69.14%	69.25%
	LET	71.95%	71.75%
	SET	68.00%	68.25%



(a) Non-defense target model

(b) Defense target model

Fig. 13. The explanation data points, which are obtained by the defense and non-defense target models, are projected into a 2D space using principal component analysis (PCA). The MNIST dataset is an example.

the same technology as Fig. 13. Thus, it can be concluded that even if the defense target model adds perturbation to the posterior probability, it has no great influence on the explanation derived from the model-agnostic XAI technology.

Discussion:

(1) Even if the target model does not provide explanation services, it still faces privacy risks caused by explanations (see Section 6.1). The adversary can obtain explanations through model-agnostic explanation technologies, and the attack accuracy is equivalent to that of the target model with explanation services.

(2) The attack accuracy of the substitution model does not increase the expensive query budget (see Section 6.2). The adversary only needs to query the target model 10 times to obtain an explanation of the specified image.

(3) The target model perturbs the confidence score provided to the user, and still can not prevent the adversary's XAI-aware model extraction attacks (see Section 6.3).

7. Related work

Our work lies in the intersection of two research fields: model explanation and model extraction attack. In this section, we give an overview separately and present the nascent work that intersects the two.

7.1. Model extraction attacks

Model extraction attacks emerge when an adversary attempts to duplicate model $\tilde{f}$ by querying the target model f. In general, there are three types of attacks for model extraction based on the adversary's objective: accuracy extraction, fidelity extraction, and functionally equivalent extraction [30]. The accuracy extraction refers to the substitution model that matches or exceeds the task accuracy of the target model. Tramèr et al. took the lead in researching accuracy extraction [31]. Since then, Orekondy et al. [32] proposed to extract the target model by reinforcement learning. It achieves the task accuracy of the substitution model beyond the target model. Chandrasekaran et al. [33] proposed to explore active learning to achieve model extraction. It reduces the number of queries with the target model and improves attack efficiency.

However, fidelity extraction requires the performance of the substitution model should not only approach the accuracy of the target model but also replicate the errors of the target model [34]. Mika et al. [16] proposed a model extraction method based on the Jacobian matrix, whose core is to extract the decision boundary of the target model. Subsequently, Zhou et al. [35] extended the work of Mika et al. [16] and proposed that the key to model decision boundary extraction is not only the generation of synthetic samples but also the selection of hyperparameters. To improve the fidelity of model extraction, Jagielski et al. [18] adopted a hybrid strategy. They theoretically derived the parameters of the first two hidden layers of the target model, and then used semi-supervised learning to train a complete substitution model.

The functional-equivalent extraction is the ultimate goal of fidelity extraction, and it is also the most difficult to realize [36]. Daniel et al. [37] proposed an efficient algorithm for reverse linear classifiers with continuous or Boolean features. Rolnick et al. [19] extracted the shallow ReLU neural network through theoretical analysis to restore the weight and structure. In addition, Batina et al. [38] and Duddu et al. [20] exploited the information leaked by the side channel.

Although all the above works have taken important steps towards model extraction attacks, their attack technologies are still limited in utilizing labels or confidence scores of the target model.

7.2. Model explanations

Although there are many model explanation technologies [21], we pay close attention to explanations of image-based neural networks. According to our knowledge of the model, it can be divided into two types: model-related and model-agnostic. Model-related techniques mean that they require knowledge of the model, such as parameters, structure, etc. Shrikumar et al. [39] proposed a decomposition method based on neural network predictions. It obtains important features through the contribution of all neurons by backpropagation. Simonyan et al. [22] proposed a gradient-based method (GRAD) to calculate the saliency map by backpropagation. Rebuffi et al. [40] conducted an in-depth analysis based on the backpropagation method and proposed NormGrad. It is a new saliency method based on the contribution of the convolution weight gradient space. Selvaraju et al. [24] proposed a technique called Grad-CAM, which uses the gradient information of the last convolutional layer to assign important values to each neuron.

However, model-agnostic techniques only need to know the prediction information output by the model. Zintgraf et al. [23] proposed a strategy to analyze the prediction difference after each input patch was marginalized. The main idea of LIME [26] is to utilize interpretability models (e.g., linear models, decision trees) to locally approximate the prediction of the target black-box model (e.g., deep neural network). It detects the change in the output of the black-box model by slightly perturbing the input image. Then it trains an interpretability model at the point of interest based on this change. Fong et al. [25] proposed another strategy (MASK) to determine attribute mapping by inputting minimal noise perturbations and observing the change in model predictions.

In this paper, we assume that when a target model does not provide explanations, the adversary can obtain the explanations corresponding to the specific input of the target model through model-agnostic technologies. This verifies that even if the target model does not actively provide the explanations, it still increases privacy risks.

7.3. Privacy risk of model explanations

Privacy threats from model explanations were first revealed in the membership inference attack [41]. The following works [17] also proved that model explanations could reduce the number of queries to the target model. As for the adversarial attack technology, the generated adversarial example could not only mislead the target model but also deceived their coupled explanation models [42]. Similarly, the model inversion attack could improve the quality of reconstructed face images by exploiting explanations [43]. However, the existing attack works only exploit model explanations for the training set and rarely extend their scope to model extraction attacks. Hence, in this paper, we investigate how to exploit different explanation types to perform model extraction attacks.

8. Conclusion

In this work, we propose an XAI-aware model extraction attack method, called XaMEA, which has three pipelines called DET, LET, and SET. XaMEA leverages the knowledge of model explanations through the idea of transfer learning. We conduct a comprehensive and detailed evaluation, and XaMEA reveals the privacy risks faced by XAI. Furthermore, to extend XaMEA to more general scenarios, we analyze the non-explainable target models through sufficient experiments. Even against non-explanation target models, the attack performance is still comparable to explanation target models. For future work, we will extend the attack for different data types (e.g., text, video, and audio) and explore a defensive technology that can trade off the transparency and privacy of the model.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 61966011, No. 62002074, and No. 62072132, the National Natural Science Foundation of China for Joint Fund Project under Grant No. U1936218, the Graduate Innovation Research Project of Hainan Province under Grant No. Qhyb2021-40.

References

[1] K. Fu, Z. Chang, Y. Zhang, G. Xu, K. Zhang, X. Sun, Rotation-aware and multi-scale convolutional neural network for object detection in remote sensing images, ISPRS J. Photogramm. Remote Sens. 161 (2020) 294–308.

[2] D. Baumann, R. Pfeffer, E. Sax, Automatic generation of critical test cases for the development of highly automated driving functions, in: 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), IEEE, 2021, pp. 1–5.

[3] E. Othman, P. Werner, F. Saxen, A. Al-Hamadi, S. Gruss, S. Walter, Automatic vs. human recognition of pain intensity from facial expression on the x-ite pain database, Sensors 21 (9) (2021) 3273.

[4] Y. He, G. Meng, K. Chen, X. Hu, J. He, DRMI: a dataset reduction technology based on mutual information for black-box attacks, in: M. Bailey, R. Greenstadt (Eds.), 30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021, USENIX Association, 2021, pp. 1901–1918.

[5] C. Wang, G. Liu, H. Huang, W. Feng, K. Peng, L. Wang, MIASec: enabling data indistinguishability against membership inference attacks in MLaaS, IEEE Trans. Sustain. Comput. 5 (3) (2020) 365–376, https://doi.org/10.1109/TSUSC.2019.2930526.

[6] H.G. Ramaswamy, et al., Ablation-cam: visual explanations for deep convolutional network via gradient-free localization, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2020, pp. 983–991.

[7] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, A. Torralba, Learning deep features for discriminative localization, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27–30, 2016, IEEE Computer Society, 2016, pp. 2921–2929.

[8] F. Hohman, H. Park, C. Robinson, D.H.P. Chau, Summit: scaling deep learning interpretability by visualizing activation and attribution summarizations, IEEE Trans. Vis. Comput. Graph. 26 (1) (2019) 1096–1106.

[9] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, D. Song, The secret revealer: generative model-inversion attacks against deep neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 253–261.

[10] C.A. Choquette-Choo, F. Tramer, N. Carlini, N. Papernot, Label-only membership inference attacks, in: International Conference on Machine Learning, PMLR, 2021, pp. 1964–1974.

[11] C.-C. Tu, P. Ting, P.-Y. Chen, S. Liu, H. Zhang, J. Yi, C.-J. Hsieh, S.-M. Cheng, AutoZOOM: autoencoder-based zeroth order optimization method for attacking black-box neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 742–749.

[12] S. Pal, Y. Gupta, A. Shukla, A. Kanade, S.K. Shevade, V. Ganapathy, ActiveThief model extraction using active learning and unannotated public data, in: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press, 2020, pp. 865–872.

[13] M. Zhou, J. Wu, Y. Liu, S. Liu, C. Zhu, DaST: data-free substitute training for adversarial attacks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020,, IEEE, 2020, pp. 231–240.

[14] Z. Yang, J. Zhang, E. Chang, Z. Liang, Neural network inversion in adversarial setting via background knowledge alignment, in: L. Cavallaro, J. Kinder, X. Wang, J. Katz (Eds.), Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019, ACM, 2019, pp. 225–240.

[15] Z. Li, Y. Zhang, Membership leakage in label-only exposures, in: Y. Kim, J. Kim, G. Vigna, E. Shi (Eds.), CCS '21: 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, Republic of Korea, November 15 - 19, 2021, ACM, 2021, pp. 880–895.

[16] M. Juuti, S. Szyller, S. Marchal, N. Asokan, PRADA: protecting against DNN model stealing attacks, in: IEEE European Symposium on Security and Privacy, EuroS&P 2019, Stockholm, Sweden, June 17-19, 2019, IEEE, 2019, pp. 512–527.

[17] S. Milli, L. Schmidt, A.D. Dragan, M. Hardt, Model reconstruction from model explanations, in: Danah Boyd, J.H. Morgenstern (Eds.), Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019, ACM, 2019, pp. 1–9.

[18] M. Jagielski, N. Carlini, D. Berthelot, A. Kurakin, N. Papernot, High accuracy and high fidelity extraction of neural networks, in: S. Capkun, F. Roesner (Eds.), 29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, USENIX Association, 2020, pp. 1345–1362.

[19] D. Rolnick, K.P. Kording, Reverse-engineering deep ReLU networks, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, in: Proceedings of Machine Learning Research, PMLR, vol. 119, 2020, pp. 8178–8187.

[20] V. Duddu, D. Samanta, D.V. Rao, V.E. Balas, Stealing neural networks via timing side channels, CoRR, arXiv:1812.11720 [abs].

[21] H. Chefer, S. Gur, L. Wolf, Transformer interpretability beyond attention visualization, CoRR, arXiv:2012.09838 [abs].

[22] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: visualising image classification models and saliency maps, in: Y. Bengio, Y. LeCun (Eds.), 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, in: Workshop Track Proceedings, 2014.

[23] L.M. Zintgraf, T.S. Cohen, T. Adel, M. Welling, Visualizing deep neural network decisions: prediction difference analysis, in: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, in: Conference Track Proceedings, 2017, OpenReview.net.

[24] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-CAM: visual explanations from deep networks via gradient-based localization, Int. J. Comput. Vis. 128 (2) (2020) 336–359, https://doi.org/10.1007/s11263-019-01228-7.

[25] R.C. Fong, A. Vedaldi, Interpretable explanations of black boxes by meaningful perturbation, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22–29, 2017, IEEE Computer Society, 2017, pp. 3449–3457.

[26] M.T. Ribeiro, S. Singh, C. Guestrin, "Why should I trust you?": explaining the predictions of any classifier, in: B. Krishnapuram, M. Shah, A.J. Smola, C.C. Aggarwal, D. Shen, R. Rastogi (Eds.), Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, ACM, 2016, pp. 1135–1144.

[27] I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, in: Conference Track Proceedings, 2015.

[28] F. Croce, M. Hein, Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks, in: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event, in: Proceedings of Machine Learning Research, PMLR, vol. 119, 2020, pp. 2206–2216.

[29] T. Orekondy, B. Schiele, M. Fritz, Prediction poisoning: towards defenses against DNN model stealing attacks, in: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, 2020, OpenReview.net.

[30] H. Hu, J. Pang, Model extraction and defenses on generative adversarial networks, arXiv preprint, arXiv:2101.02069.

[31] F. Tramèr, F. Zhang, A. Juels, M.K. Reiter, T. Ristenpart, Stealing machine learning models via prediction APIs, in: 25th {USENIX} Security Symposium ({USENIX} Security 16), 2016, pp. 601–618.

[32] T. Orekondy, B. Schiele, M. Fritz, Knockoff nets: stealing functionality of black-box models, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019, Computer Vision Foundation/IEEE, 2019, pp. 4954–4963.

[33] V. Chandrasekaran, K. Chaudhuri, I. Giacomelli, S. Jha, S. Yan, Exploring connections between active learning and model extraction, in: S. Capkun, F. Roesner (Eds.), 29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, USENIX Association, 2020, pp. 1309–1326.

[34] J.R. Correia-Silva, R.F. Berriel, C. Badue, A.F. de Souza, T. Oliveira-Santos, Copycat CNN: stealing knowledge by persuading confession with random non-labeled data, in: 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1–8.

[35] M. Zhou, J. Wu, Y. Liu, S. Liu, C. Zhu, DaST: data-free substitute training for adversarial attacks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 234–243.

[36] Y. Zhu, Y. Cheng, H. Zhou, Y. Lu, Hermes attack: steal {DNN} models with lossless inference accuracy, in: 30th {USENIX} Security Symposium ({USENIX} Security 21), 2021.

[37] D. Lowd, C. Meek, Adversarial learning, in: Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, 2005, pp. 641–647.

[38] L. Batina, S. Bhasin, D. Jap, S. Picek, CSI neural network: using side-channels to recover your artificial neural network information, IACR Cryptol. ePrint Arch. 2018 (2018) 477.

[39] A. Shrikumar, P. Greenside, A. Kundaje, Learning important features through propagating activation differences, in: D. Precup, Y.W. Teh (Eds.), Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6–11 August 2017, in: Proceedings of Machine Learning Research, PMLR, vol. 70, 2017, pp. 3145–3153.

[40] S. Rebuffi, R. Fong, X. Ji, A. Vedaldi, There and back again: revisiting backpropagation saliency methods, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, IEEE, 2020, pp. 8836–8845.

[41] R. Shokri, M. Strobel, Y. Zick, On the privacy risks of model explanations, arXiv preprint, arXiv:1907.00164.

[42] X. Zhang, N. Wang, H. Shen, S. Ji, X. Luo, T. Wang, Interpretable deep learning under fire, in: S. Capkun, F. Roesner (Eds.), 29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020, USENIX Association, 2020, pp. 1659–1676.

[43] X. Zhao, W. Zhang, X. Xiao, B.Y. Lim, Exploiting explanations for model inversion attacks, CoRR, arXiv:2104.12669 [abs].