Black-box based limited query membership inference attack

YU ZHANG, HUAPING ZHOU, PENGYAN WANG, AND GAOMING YANG
School of Computer Science and Engineering, Anhui University of Science and Technology, Huainan 232001, China (e-mail: yuzhang@aust.edu.cn)
Corresponding author: Yu Zhang (e-mail: yuzhang@aust.edu.cn).

This work is supported by the Natural Science Foundation of Anhui Province under Grant 2008085MF220; the Natural Science Foundation of Universities in Anhui Province under Grant KJ2019A0109 and KJ2018A0084.

ABSTRACT Conventional membership inference attacks usually require a large number of queries of the target model when training shadow models, and this task becomes extremely difficult when the number of queries is limited. Aiming at the problem of insufficient training data for shadow models due to the limited number of queries, we propose a membership inference attack method based on generative adversarial networks (GAN). First, we use generative adversarial networks to augment the samples obtained by a small number of queries to expand the training data of the model; Secondly, we use the improved CNN to obtain shadow models that have a higher degree of fitting on different target model structures; Finally, we evaluate the accuracy of the proposed algorithm on XgBoost, Logistic, and neural network models using public datasets MNIST and CIFAR10 in a black-box setting, and the results show that our model has an average attack accuracy of 62% and 83%, respectively. It can be seen that, compared with the existing research methods, our model can obtain better attack effects under the condition of significantly reducing the number of queries, which shows the feasibility of our proposed method in membership inference attacks.

INDEX TERMS membership inference attack; generative adversarial network; black-box attack; information leak

I. INTRODUCTION

WITH the great progress of big data processing technology and hardware computing power, machine learning (ML) has developed rapidly in many fields such as image and speech recognition, autonomous driving[5], network security, sentiment analysis, etc[1–4]. However, While machine learning brings convenience to people’s lives, it also brings severe challenges to data privacy and security, such as adversarial sample attacks[1], attribute attacks[2] and inference attacks[25]. This is mainly because machine learning models require a large amount of training data, and these data inevitably contain users’ private information. If such privacy is leaked, it will cause serious harm to users. For example, assuming that there is a model trained on cancer patient information, when an attacker knows that someone’s data is a member of the model’s training set, he can directly infer that person’s health status and thus may lead to discrimination problems. This kind of attack to infer whether the data is in the training dataset is called membership inference attack[10, 16, 27]. For defenders, only if they fully understand the attacker’s means can they take targeted defense measures to protect user privacy. It can be seen that the research on membership inference attacks is conducive to improving the security of machine learning algorithms.

Membership inference attacks determine whether a certain piece of data exists in the training data set of the target model by distinguishing the performance difference between the machine learning model on the training set and the non-training set[26]. Given that in the black-box situation, the attacker cannot obtain this discrimination ability, Shokri et al.[6] propose to train a set of ‘shadow models’ that simulate the behavior of the target model to solve this problem. First, they synthesize data similar to the training set of the target model by querying the target model; Second, they used synthetic data to construct a series of ‘shadow models’ that mimicked the behavior of the target model. Compared with the target model, whether the data is a member of the training set, i.e. the data membership of the shadow model is known; Finally, they combined the membership information with the input and output of the shadow model to form a dataset, and trained the attack model to distinguish the behavior differences of the target model on different datasets, so as to realize the membership inference attack. Although the
basic idea does not sound complicated, it is not easy for an attacker to obtain a sufficiently similar shadow model, which is manifested in the following two aspects:

1) Insufficient training data. Training shadow models requires a lot of training data. However, existing data synthesis strategies often require a large number of queries to obtain a single data model. For example, the synthetic data generation strategy of Shokri et al.[6], even on datasets with binary features, synthesizing a single data point requires 156 queries to the target model[7]. In fact, due to the query cost and model defense, etc., it is difficult to achieve frequent queries to the target model. The small amount of data obtained under restricted query is not enough to train a shadow model[7] that is completely similar to the target model.

2) Poor adaptability. In the case of a black box with unknown information such as the target model structure, model parameters and training data set, it is difficult for an attacker to obtain a highly adaptive shadow model with good fitting ability to various target model structures[8, 9]. Therefore, to address these two problems, we propose a query-restricted membership inference attack strategy. When limiting the number of query target models, this strategy can ensure that membership inference attacks can be carried out effectively on different target models.

The contributions of this work mainly include three aspects:

- In order to deal with the shortage of shadow model training data caused by the limited number of queries, we propose a method to enhance the shadow model training data. We utilize a small number of data samples obtained by querying the target model and generate new samples through generative adversarial networks[10] to augment the training data with synthetic samples. This can speed up the training process of membership inference attacks without additional calls to the target model.

- Since it is difficult to obtain a machine learning model with similar functions to the target model when the access to the target model is limited, we use a deep neural network to design a shadow model that simulates the prediction function of the target model. The target models can achieve better results and have better adaptability.

- To verify the effectiveness and generality of the model, we use the accuracy rate to quantify the performance of the attack model and conduct experiments on classic machine learning models (Logistic, XgBoost and neural networks). Compared with Shokri’s experiments, our method can achieve higher attack accuracy under the same conditions.

The rest of this paper is organized as follows. The first part introduces the relevant knowledge; the second part elaborates the designed attack method; the third part is the experimental evaluation; finally the fourth part summarizes the full text.

## II. RELATED WORK

### A. MEMBERSHIP INFEERENCE ATTACK

Since Shokri et al. [6] proposed an inference attack against MLaas, the research on member inference has received extensive attention and has been successfully implemented in many fields. It uses the output information of the model to infer whether a sample exists in the training data set of the model, which seriously threatens the privacy and security of the machine learning model. Membership inference attacks are divided into white-box attacks[11–14] and black-box attacks[5, 6, 15] according to background knowledge. In the case of a black-box attack, the attacker does not know the structure and network of the target model, and can only predict the result of the input data by interacting with the machine learning algorithm. On the contrary, the attacker can fully access all the structure and parameters of the target model under the white box attack, which has a very strong attack capability. In addition, membership inference attacks can also be divided into attacks against independent models[16–21] and attacks against Federated Learning[11, 22].

At present, the research on membership inference attack mainly focuses on the black-box attack of independent models. The attacker uses the model’s predicted behavior difference between member data and non-member data to train a binary attack model that recognizes this difference. In order to obtain the training data of the attacking model in the black-box mode, Shokri[6] uses a synthetic dataset to train on a model with a similar structure to the target model. They first obtain k shadow models that simulate the output of the target model. Second, they use the training set and non-training data of the shadow model to query the target model respectively, and mark the obtained probability vectors as 'in' and 'out'; on this basis, they use these labeled data to train the attack model. Finally, given a piece of data to be predicted, the attack model predicts whether it is a member of the training data set according to the output of the data on the target model. The training process of membership inference attack is shown in Figure 1.

Shokri’s method[6] is based on the assumption of extensive query and knowledge of the target model structure. They obtained synthetic data by querying the target model and used this data to train multiple shadow models with the same structure as the target model. In practical applications, a large number of accesses to the target model require high query costs, which largely limits the application scope of membership inference attacks. Therefore, how to construct a shadow model similar to the target model under a small number of queries is the key to solving the hypothesis of membership inference attack.

### B. GENERATIVE ADVERSARIAL NETWORKS

Since small datasets usually cannot provide enough representative shadow training samples, this may reduce the similarity between the shadow model and the target model, which in turn affects the accuracy of membership inference attacks. Effective data augmentation methods[28] help reduce the
number of queries to the model and improve the fit of shadow models, so data augmentation is necessary to apply shadow model techniques to membership inference attacks with small datasets.

Generative adversarial network was first proposed by Goodfellow et al. [10] in 2014, and it is a powerful generative model composed of two neural networks that play against each other. It learns the data distribution of training samples and generates synthetic samples that are similar to the input data. In addition, GAN-based data augmentation is not limited by the type and size of training data, and a small amount of data can generate data with high realism and diversity. However, traditional generative adversarial networks cannot allow users to synthesize samples that meet their own needs, and it is easy to cause instability in training or model collapse. Based on this, Marza et al. [23] proposed the CGAN model, which is an effective improvement on the original GAN. By adding the same labels in the generator and discriminator, it guides the generator to generate samples, speeds up training and solves the model Training instability problem.

Clearly, conditional generative adversarial networks offer the possibility to stably synthesize a large number of shadow model training samples. Therefore, this paper combines CGAN and shadow model techniques to conduct membership inference attacks. We generate more similar data samples from the original training dataset through CGAN to participate in the training of the shadow model, improve the similarity between the shadow model and the target model, and thus improve the performance of inference attacks.

III. DATA AUGMENTED SHADOW MODEL TRAINING

This section describes how to implement membership inference attack with limited number of queries. The method consists of data augmentation and shadow model training. In order to allow the shadow model to better simulate the function of the target model when the training data is limited, we propose to use GAN to enhance the training data. Different from existing shadow training methods, this paper uses a deep neural network to learn the prediction distribution of the target model.

Sufficient training data can train a shadow model that is more functionally similar to the target model than with a small amount of data. We can rely on the shadow model to capture the functional difference between the member data and non-member data of the target model, and obtain better attack performance. Based on this idea, we combine CGAN and shadow model for data augmentation, and finally achieve membership inference attack with limited number of queries. Since the CGAN model adds additional constraints on the basis of GAN to control the network output, we first analyze the implementation of the GAN framework, and then introduce the specific implementation of CGAN. Specifically, we use GAN to enhance the small amount of data obtained in the case of limited query to increase the quantity and diversity of shadow model training data. In this way, the fit of the model to the target model can be improved, and the attack model can distinguish between member and non-member data. As shown in Figure 2, the attack process is mainly divided into three stages.

FIGURE 1: System structure of membership inference attacks
A. GAN-ENHANCED SYNTHETIC DATA

Figure 2 is the overall framework of the membership inference attack. We augment synthetic datasets with GAN[10] to better train shadow models without additional queries, which is also an extension to the application scenario of membership inference attacks in Section II. Compared with traditional neural network-based data augmentation methods such as VAE[29] and PixelRNN[30], GAN[31] has the advantages of higher quality synthetic data, more similar to the original data, and can be synthesized in large quantities in a short period of time, which is exactly what is needed for training shadow models.

As shown in Figure 3, GAN consists of two arbitrarily differentiable functions, one part is the generator G and the other part is the discriminator D. The generator G accepts random noise z and outputs synthetic data G(z); the discriminator D receives real data or synthetic data, and makes a binary judgment on the data. The goal of D is to make D(x) close to 1 when the input data is sampled from the real data x, whereas D(G(z)) is close to 0 if the data is sampled from the generated data G(z). For G, the goal is to generate augmented data G(z) with a similar distribution to the real data, so that the performance of G(z) on D is consistent with the performance of real data x on D. This is actually a Mini-Max game between the two, and the performance of G and D is improved in the continuous confrontation competition. When D cannot judge the source of the data, it is considered that G has learned the distribution of the original data. However, the GAN training in Figure 3 is too liberal and cannot generate class-specific data. In order to solve this problem, this paper uses the CGAN[23] structure after comprehensive consideration.

CGAN is an extended conditional constrained generative adversarial model to the original GAN. We control the output of the network by adding additional constraints y during the training of the generator and discriminator. The constraints can be class labels or data of different modalities. In order to specify the class of the generated data, the class labels are used as constraints to input into the generation network and the discriminant network.

Figure 4 is the overall framework of CGAN for enhancing synthetic data. The framework adopts an alternate optimization method. First we fix G, replace synthetic data with real data of D, and train D with real data and constraint y. Adjust the parameters of D until D obtains the best judgment accuracy on real data and generated data. Then, we fix the parameters of D and train G. We will form a joint hidden layer expression input G, which conforms to the normal distribution noise z and constraint y, and optimize the parameters of G to minimize the data generated by G as much as possible. Judgment accuracy. Finally, we can get a real data generator. The optimization goal is:

\[
\min_{G} \max_{D} V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log D(x) \right] - \mathbb{E}_{z \sim p_z} \left[ \log (1 - D(G(z))) \right].
\]

In the data augmentation stage, we take x_{syn} obtained by querying as the original dataset, and use CGAN to generate a synthetic dataset \( D_{\text{train}}(x_{gan}, y_{gan}) \) with specified labels to enrich the original training set of the shadow model. That is, \( x_{syn} \) and \( x_{gan} \) form a new training set \( x_{aug}(x_{aug} = x_{syn} \cup x_{gan}) \). To make the shadow model function as similar as possible to the target model, we first process the augmented dataset \( x_{aug} \). We divide it into two parts: one part consists of \( x_{aug}^{inc} \) which is relatively similar to the target model training set; the other part consists of \( x_{aug}^{out} \) which is not similar to the target model training data.

We first obtain the label probability vector for all data records in the augmented dataset \( x_{gan} \) by querying the target model \( T \):

\[
T(x_{gan}) = (p_{1}, p_{2}, \ldots, p_{k}),
\]

where \( p_{1} + p_{2} + \ldots + p_{k} = 1 \) (a total of \( k \) classification labels). Then, we select the probability value \( T'(x_{gan}) \) at the corresponding position \( i \) of the category \( y_{gan} \) of each piece of data in \( x_{gan} \). Since machine learning often behaves differently on training set data than it sees ‘first time’, membership inference attacks exploit this feature, looking for data that give a high probability value of \( T'(x) \). So when the \( T'(x_{gan}) \) of a piece of data is greater than the threshold, we consider it to be more similar to the training data. This part of the data larger than the threshold is combined with \( x_{syn} \), and the synthetic data \( x_{aug}^{inc} \) similar to the target model training data is obtained, and this part will be used to train the shadow model. The remaining synthetic data smaller than the threshold is marked as \( x_{aug}^{out} \) and used for the test set of the shadow model.

B. TRAIN THE SHADOW MODEL

For black-box attacks, it largely depends on the similarity between the shadow model and the target model, that is, the higher the similarity, the easier it is to use the known membership relationship to find the prediction difference between the training data and the test data of the target model[18]. Our goal wants to generate a machine learning model with almost the same predictive behavior as the target model. So after the first step is over, the attacker gets an augmented dataset that approximates the training data of the target model, and generates a shadow model on this dataset. The attacker can then achieve an attack on the target model.

In the selection of shadow model structure, this paper uses neural network to simulate the target model[8, 9], because the structure, parameters and other information of the target model cannot be obtained. The network defines two constraint functions (category constraint and distance constraint), so that the shadow model has a good prediction accuracy on the simulated training set. The convolutional neural network loss function is expressed as follows:

\[
L = CE(S(x_{aug}^{in}), y_{aug}^{in}) + \alpha \ast d(S(x_{aug}^{in}), T(x_{aug}^{in}))
\]

where, \( S(x_{aug}^{in}) \) and \( T(x_{aug}^{in}) \) are the output probabilities of \( x_{aug}^{in} \) on the shadow model and the target model, respectively. \( y_{aug}^{in} \) is the ground-truth class of \( x_{aug}^{in} \), and the a parameter is
the weight that controls the distance loss (in the experiment, it is set to 0.2).

The first part \( CE(S(x_{aug}^{in}),y_{aug}^{in}) \) of Equation 3 is the label class constraint, which represents the cross-entropy loss between the probability vector and the data label; the label class constraint ensures that the shadow model makes a correct judgment on the input data, that is, the probability value obtained by the shadow model on the target class maximum. The second part of Equation 3, \( d(S(x_{aug}^{in}),T(x_{aug}^{in})) \), is the distance constraint, which represents the distance index between the output of the shadow model and the target model, that is, the distance constraint is used to guide the prediction behavior of \( S \) imitating \( T \). The metric formula for distance is expressed as follows:

\[
d(S(x_{aug}^{in}),T(x_{aug}^{in})) = MSE(S(x_{aug}^{in}),T(x_{aug}^{in})). \tag{4}
\]

C. Train the Attack Model

In order to generate the attack model, this paper labels the prediction output of the shadow model, the class label and the membership state of its members, and composes them into a new attack model training dataset. We use this training set to train a shadow model so that the model can effectively distinguish training dataset members from non-training set members.

Figure 5 details how the attack model \( F_{attack} \) is trained. In the first step, according to the existing training dataset \( (x_{aug}^{in},y_{aug}^{in}) \) and test dataset \( (x_{aug}^{out},y_{aug}^{out}) \), we obtain the predicted probabilities \( S(x_{aug}^{in}) \) and \( S(x_{aug}^{out}) \) of all data records in and by querying the shadow model. Since the shadow model is trained on \( x_{aug}^{in} \), the \( S(x_{aug}^{in}) \) is marked 'in' and the rest are marked 'out'. This way the attacker has the synthetic data \( (x_{aug}^{in},y_{aug}^{in}) \), the output \( S(x_{aug}^{in}) \) of the shadow model, and the 'in/out' flags. In the second step, we add \( (y_{aug}^{in},S(x_{aug}^{in}),in) \) and \( (y_{aug}^{out},S(x_{aug}^{out}),out) \) to the training set \( D_{train\_attack} \) of the attack model. In the third step, we divide \( D_{train\_attack} \) into \( k \) parts according to different categories, and use the data in each part to supervise training to obtain a separate attack model \( F_{attack}^{k} \). The attacker takes the output vector of specific data on the target model as input, and selects the attack model according to the category to predict its member state.

IV. Experimental Results and Their Analysis

This section elaborates the details of the experiments, including dataset information, experimental environment, and model parameters. At the same time, we also present the evaluation results of experiments under different model settings, and the impact of overfitting and shadow model training dataset size on membership inference attacks.

A. Experimental Environment and Dataset

Our attack experiments are performed on a PyTorch deep learning workstation, which is equipped with two NVIDIA GeForce 1080Ti GPU cards with 11GB memory. We use
two public datasets, MNIST and CIFAR10. MNIST is a grayscale image dataset of handwritten digits with 10 categories, containing 60,000 training images and 10,000 test image datasets.\(^1\) The size of each picture is 32*32, and the digital image is located in the center of the picture after normalization. The CIFAR10 dataset contains 10 categories of color images, each of which is also 32*32 in size and consists of 60,000 images, including 50,000 training images and 10,000 test images.\(^2\) In addition, the CIFAR10 dataset contains real objects in the real world, which is more complex than the MNIST dataset. Both datasets have been used to evaluate the membership inference attack of Shokri et al.[6].

**B. EXPERIMENTAL SETUP**

1) The target model
   In order to verify the impact of different types of target models on the attack model accuracy of membership inference, this paper selects three different types of classification models with the same training set size. In traditional machine learning algorithms, we choose Logistic regression and XgBoost as target models. For deep learning based algorithms, we choose convolutional neural network as the target model. In addition, all attacks against the target model are carried out under the black-box assumption, and the attacker only has the ability to access the target model and obtain the output results of the model.

2) Shadow model
   Different from the attack method of Shokri et al.[6], we do not train multiple shadow models (one for each class), and only uses a multi-layer neural network as the shadow model to complete the simulation of the target model. This strategy greatly reduces the computational overhead of performing membership inference. In the experiments, we use the PyTorch library to build a shadow model of the convolutional neural network structure. The model takes 32*32 images as input, and the specific structure is shown in Figure 6.

3) Attack model
   The attack model uses the output results of specific data on the target model to determine whether it exists in the training set of the model, which is essentially a standard binary classification problem. So we can use any machine learning framework with binary classification capabilities to generate attack models. Here, we choose Logistic Regression Model, Random Forest and XgBoost as the binary classifier. All experiments were run 10 times with 10-fold cross-validation, and the average value was taken as the final result.

**C. EVALUATION CRITERIA**

The purpose of the membership inference attack is to infer whether a data record is in the training set of the target model. In the evaluation experiments, we first set the membership dataset to be equal in size to the non-member dataset, so that the baseline accuracy of random guessing is 0.5. Meanwhile, to verify the effectiveness of our method, we use the accuracy to evaluate the classification results of attacking models on target models with different structures. The accuracy rate here refers to the ratio of the number of data correctly classified by the attack model to the total number of test data for a given test set. The calculation formula is:

\[
\text{Accuracy}_i = \frac{N'_i}{N_i},
\]

where, \(i\) is the category of data, \(\text{Accuracy}_i\) is the accuracy of attacking model \(i\), \(N'_i\) is the total number of data with category \(i\), and \(N_i\) is the number of pairs of data. Generally speaking, the higher the accuracy, the better the attack model will perform.

---

\(^1\) http://yann.lecun.com/exdb/mnist/
\(^2\) http://www.cs.toronto.edu/~kriz/cifar.html
D. PERFORMANCE OF MEMBER ATTACK UNDER GAN ENHANCEMENT

Based on Pytorch, we design a deep learning network for generating CGANs. The generator and discriminator consisting of neural networks have the following parts:

The generator $G$ takes a 100-dimensional noise vector $z$ and a class label $c$ conforming to the normal distribution as input, and outputs the augmented data $G(z|c)$ of the same dimension as the original data. The generator $G$ contains 3 fully connected layers, which are connected to the next layer through the LeRU activation function. This activation function has the advantages of ReLU and avoids the vanishing gradient problem of neural networks. The final output layer uses a hyperbolic tangent ($\tanh$) activation function. Where $\text{Lin}$ refers to the linear layer, and LeRU refers to the leakage correction linear activation function LeakyReLU. Compared with the ReLU activation function, the LeRU function can expand the processing range. Values on some negative axes that are less than 0 can also be well preserved, so that the information on the negative axis will not be lost.

The discriminator $D$ accepts the original 32*32-dimensional data generated by the generator $G$, and makes a binary judgment on the data to determine whether it comes from the original real data or the fake data generated by the generator $G$. The network structure of the discriminator $D$ is similar to that of the generator $G$. The neurons of his 3 fully connected layers are activated by LeRU, the difference is that the output of the activation function goes through the Dropout layer before connecting to the next layer, and the final output layer uses the Sigmoid activation function.

In the case of few queries, we use CGAN to generate labeled synthetic data to augment the training dataset of the shadow model. CGAN is trained with 500 epochs with a batch size of 32 samples. The loss functions of the generator and discriminator during training are shown in Figure 7. We first increased the number of samples in each category from the original 200 to 1000, and then randomly divided the training set and the test set with the enhanced 6000 shadow model training data according to the ratio of 3/4. Finally, when training the attack model, we randomly select 3/4 of the shadow model training data to be labeled with ‘in’, and the rest are labeled with ‘out’. For the test set, we label the dataset in the same way.

Table 1 and Table 2 are the attack accuracy of different target models CNN, XgBoost, and Logistic based on MNIST and CIFAR10 datasets. At the same time, in order to evaluate the impact of the attack model structure on the membership inference attack performance, we tested the attack models trained by different machine learning models, including the attack effects of Logistic, XgBoost, and RandomForest. It can be seen from the columns in Table 1 and Table 2 that for the CIFAR10 and MNIST datasets, when the target model is determined, the change of the attack model has relatively little impact on the member’s inferred attack performance.

Each row in Table 1 is the attack accuracy on different target models, and the attack accuracy of all target models is 0.5 higher than the baseline. It can be seen from Table 1 that the attacks against the CNN and XgBoost target models have
similar accuracy, and when the target model adopts the CNN model, the average attack accuracy can reach 0.64. For the XgBoost model, the CNN model achieved an attack success rate of 0.63, and the Logistic model had the worst attack performance with an average attack success rate of 0.59, which was 4% lower than the XgBoost performance and 5% lower than the CNN attack performance. This is mainly because the Logistic algorithm has the worst performance on the classification task compared with the other two algorithms, and the classification accuracy affects the inferred attack accuracy.

For the CIFAR10 dataset, Table 2 is the highest attack performance obtained on the target model CNN, which achieves an average attack accuracy of 0.86 on three different structured attack models. For the target models trained with XgBoost, the attack success rate is above 80%. Even on the worst logistic model, you can still get an average precision of 0.77, which is much better than random guessing. The experimental results show that our method can effectively infer the membership degrees of the training data of different target models, and the attack effect has nothing to do with the attack model.

Figure 8 shows the attack accuracy obtained for each category on the MNIST and CIFAR10 datasets when the target model is XgBoost and the attack model is random forest. Obviously, our attack method has high attack accuracy on most classes of MNIST dataset and CIFAR10 dataset, however, it has lower accuracy on a few classes, such as the class labels of MNIST dataset are 3 and 9. The class labels of the CIFAR10 dataset are 2, 7 and 9. This is due to the different quality of data generated by GANs for different categories. The difference in the quality of the generated data will affect the fitting degree of the shadow model on different categories, and the final attack accuracy will also be different. So how to control the quality of the data generated for each category is worth discussing next.

In order to illustrate the advancement and superiority of our method, we train the attack model according to the attack experimental environment configuration of Shokri et al. [6], and compare the performance of Shokri et al.’s attack method on the MNIST and CIFAR10 datasets in terms of attack accuracy. The results are shown in Table 3.

In Table 3, our method greatly improves the accuracy of the attack while reducing the number of times of querying the target model. Compared with Shokri’s scheme, our shadow model samples 5x more training data for CNN, XgBoost and Logistic models on the MNIST dataset, and improves the attack accuracy by about 15%, even on the more complex CIFAR10 dataset, the attack performance still a slight improvement (about 5%). The experimental results show that our attack has a higher attack accuracy than the previous work.

### E. IMPACT OF SHADOW TRAINING DATA

This section evaluates the impact of the size of the shadow model dataset on the performance of the attacked model. To illustrate the relationship between shadow model dataset size and membership inference, we use the ResNet network model to train a set of shadow models on two shadow datasets of different sizes. Under the condition that the target model and attack model remain unchanged, the experiments are reevaluated, and the experimental results are shown in Figure 9. It is clear that the attack accuracy on the MNIST dataset and the CIFAR10 dataset increases with the size of the shadow dataset. When the shadow training data size of each class in the MNIST dataset is 1000, the overall attack accuracy improves from 53.7% to 61.8%. For the CIFAR10 dataset, the shadow dataset size has less impact on the attack. When the number of training data for each class is 200, the overall attack accuracy is 77.8%. However, when the training data per class is increased to 1000, the accuracy of the attack increases to 82.3%, an improvement of 4.5%.

Therefore, it can be seen that the membership inference attack accuracy improves with the increase of shadow model training data. This is because when more training data is used to query the target model, the attacker can obtain more information about the training/testing data through the prediction vector, which will enhance the fitting of the shadow model, thereby improving the accuracy of the attacking model.

### F. INFLUENCE OF OVERFITTING OF TARGET MODEL ON ATTACK

This section evaluates the impact of overfitting on membership inference attacks. In order to obtain two target models with different fitting degrees, keeping variables such as model parameters and structure unchanged, we only modify

| Target model | CNN | XgBoost | Logistic |
|--------------|-----|---------|----------|
| LR           | 0.86| 0.85    | 0.76     |
| XB           | 0.88| 0.82    | 0.79     |
| RF           | 0.86| 0.84    | 0.78     |

**TABLE 2: Accuracy on membership inference for different target and attack models on CIFAR10**

![Figure 8: Attack accuracy against XgBoost](image)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3175824, IEEE Access

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
the size of the model training set. Figure 10 shows the performance of the attacker before and after overfitting the target model. It can be clearly seen that when attacking MNIST, the average attack accuracy rate of 53.1% can be obtained on the target model trained with 10,000 data, while the average attack accuracy on the overfit model with only 1,000 training data can be obtained. reached 59.3%. That said, CIFAR10 is more vulnerable to membership inference attacks than MNIST-trained models. On the overfit target model, the membership inference attack achieves 88% accuracy.

Experimental results show that, compared with machine learning models with good generalization ability, overfitting models are more vulnerable to membership inference attacks. The behavior of the overfitting model on the training data set and the non-training data set is quite different, that is, the prediction output of the member data is more concentrated than the prediction output of the non-member data. Therefore, it is easier for the attacked model to learn how to distinguish this difference and obtain higher attack accuracy.

V. CONCLUSION
Membership inference attacks based on shadow models often require a large amount of training data. This is based on extensive querying of the target model, which not only consumes a lot of computer resources and time, but is also easily detected by defenders. To address this problem, we propose a query-constrained data-augmented membership inference attack. First, we use generative adversarial networks to enhance the dataset obtained under a small number of queries; Then, we use class constraints and distance constraints to train a deep neural network to learn the prediction behavior of the target model, thereby improving the fit and adaptability of the shadow model to the target model. Finally, we use information such as membership to train a binary classification attack model. Using comparative experiments, it can be found that the attack accuracy of our method is higher than that of Shokri’s membership inference attack method when the number of queries is limited, and the assumption that training the shadow model requires a large number of queries of the target model is relaxed. However, our proposed data-augmented membership inference attack algorithm still has a lot of room for improvement. At present, we only consider attacks on small-scale datasets, and on large-scale datasets such as ImageNet, CelebA, etc., doing so cannot achieve high attack accuracy. Therefore, our future work will try to design a more reasonable generative adversarial network, optimize the shadow model structure, and increase the network depth to improve the attack performance on large-scale datasets.

REFERENCES
[1] K. Ren, T. Zheng, Z. Qin, and X. Liu, “Adversarial attacks and defenses in deep learning,” Engineering, vol. 6, no. 3, pp. 346–360, May 2020, 10.1016/j.eng.2019.12.012.
[2] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, “When machine unlearning jeopardizes privacy,” in Proc. CCS’21, New York, NY, USA, 2021, pp. 896–911.
[3] M. A. Rahman, T. Rahman, R. Laganière, N. Mohammed, and Y. Wang, “Membership inference attack against differentially private deep learning model,” Trans. Data Priv., vol. 11, no. 1, pp. 61–79, Feb. 2018.
[4] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. CVPR*, Las Vegas, NV, USA, 2016, pp. 770–778.

[5] E. Yurtsever, J. Lamabert, A. Carballo, and K. Takeda, “A survey of autonomous driving: Common practices and emerging technologies,” *IEEE Access*, vol. 8, pp. 58443–58469, Mar. 2020, 10.1109/ACCESS.2020.2983149.

[6] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, “Membership inference attacks against machine learning models,” in *S&P*, San Jose, CA, USA, 2017, pp. 3–18.

[7] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, “MI-Leaks: Model and data independent membership inference attacks and defenses on machine learning models,” in *NDSS*, Reston, VA, USA, 2019, pp. 24–27.

[8] Y. Shi, Y. E. Sagduyu, K. Davaslioglu, and J. H. Li, “Generative adversarial networks for black-box api attacks with limited training data,” in *ISSPIT*, Louisville, KY, USA, 2018, pp. 453–458.

[9] F Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, “Stealing machine learning models via prediction apis,” in *Proc. USENIX*, 25th ed., Austin, TX, USA, 2016, pp. 601–618.

[10] G. Y. Liu, C. Wang, K. Peng, H. Huang, Y. Li, and W. Cheng, “Socinf: Membership inference attacks on social media health data with machine learning,” *IEEE Trans. Comput. Soc. Syst.*, vol. 6, no. 5, pp. 907–921, Jun. 2019.

[11] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” *arXiv preprint arXiv:1411.1784*, 2014.

[12] M. Nasr, R. Shokri, and A. Houmansadr, “Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning” in *S&P*, San Francisco, CA, USA, 2019, pp. 739–753.

[13] K. Leino and M. Fredriksson, “Stolen memories: Leveraging model memorization for calibrated white-box membership inference,” in *Proc. USENIX*, 29th ed., Boston, MA, USA, 2020, pp. 1605–1622.

[14] A. Hayes, L. Melis, G. Danezis, and E. D. Cristofaro, “Logan: Membership inference attacks against generative models,” *Proc. Priv. Enhancing Technol.*, Sep. 2019, pp. 133–152.

[15] D. Chen, N. Yu, Y. Zhang, and M. Fritz, “Gan-leaks: A taxonomy of membership inference attacks against generative models,” in *Proc. CCS’20*, New York, NY, USA, 2020, pp. 343–362.

[16] U. Gupta, D. Stripelis, P. K. Lam, P. Thompson, J. L. Ambite, and G. V. Steeg, “Membership inference attacks on deep regression models for neuroimaging,” *PMLR*, vol. 143, pp. 228–251, Jul. 2021.

[17] B. Hilprecht, M. Härricher, and D. Bernau, “Monte carlo and reconstruction membership inference attacks against generative models,” *PETS*, vol. 4, pp. 232–249, Jun. 2019.

[18] C. A. Choquette-Choo, F. Tramer, N. Carlini, and N. Papernot, “Label-only membership inference attacks,” in *PMLR*, vol. 139, pp. 1964-1974, Jul. 2020.

[19] L. Zheng and Z. Yang, “Label-Leaks: Membership inference attack with label,” *arXiv preprint arXiv:2007.15528*, Aug. 2020.

[20] P. Irolla and G. Châtel, “Demystifying the membership inference attack,” in *CMU*, 12th ed., Copenhagen, Denmark, 2019, pp. 1-7.

[21] S. Truex, L. Liu, M. E. Gursoy, L. Yu, and W. Wei, “Demystifying membership inference attacks in machine learning as a service,” *IEEE T. Serv. Comput.*, vol. 14, no. 6, pp. 2073–2089, 2019, 10.1109/TSC.2019.2897554.

[22] J. C. Li, N. Li, and B. Ribeiro, “Membership inference attacks and defenses in supervised learning via generalization gap,” *arXiv preprint arXiv:2002.12062*, Mar.
Yu Zhang et al.: Black-box based limited query membership inference attack

2020.

[23] J. Zhang, J. Zhang, J. Chen, and S. Yu, “Gan enhanced membership inference: A passive local attack in federated learning,” in ICC, Dublin, Ireland, 2020, pp. 7–11.

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” NIPS, Montreal, Quebec, Canada, 2014, pp. 2672–2680.

[25] D. Pasquini, G. Ateniese, and M. Bernaschi, “Unleashing the tiger: Inference attacks on split learning,” in Proc. CCS’21, NewYork, NY, USA, 2021, pp.2113–2129.

[26] J. Jia, A. Salem, M. Backes, Y. Zhang, and N. Z. Gong, “Memguard: Defending against black-box membership inference attacks via adversarial examples,” in Proc. CCS’19, in London, UK, 2019, pp.259–274.

[27] X. He and Y. Zhang, “Quantifying and mitigating privacy risks of contrastive learning,” in Proc. CCS’21, NewYork, NY, USA, 2021, pp.845–863.

[28] A. Mikolajczyk and M. Grochowski, “Data augmentation for improving deep learning in image classification problem,” in IEEE IIPhDW, winou,cie, Poland, 2018, pp. 117–122.

[29] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, “Autoencoding beyond pixels using a learned similarity metric,” in ICML, New York, NY, USA, 2015, pp.1558–1566.

[30] A. V. Oord , N. Kalchbrenner, and K. Kavukcuoglu, “Pixel recurrent neural networks,” in PMLR, vol.48, pp.1747–1756, Jun. 2016.

[31] A. Hu, R. Xie, Z. Lu, A. Hu, and M. Xue, “TableGAN-MCA: Evaluating Membership Collisions of GAN-Synthesized Tabular Data Releasing,” in Proc. CCS’21, NewYork, NY, USA, 2021, pp. 2096-2112.

HUAPING ZHOU was born in Tanghe, Henan Province, China in 1979. She received the M.S. degrees in power electronics from the Anhui University of Science and Technology, in 2004 and the Ph.D. degree in computer application technology from China University of Mining, Beijing, in 2012. She is currently a Professor and a M.S. Supervisor with the Anhui University of Science and Technology. And her research interests include machine learning, target detection.

PENGYAN WANG received the M.S. degree in computer technology from the Anhui University of Science and Technology, Huainan, China, in 2022. His research interests include deep learning, AI security and privacy preserving.

GAOMING YANG received the M.S. degree in computer application from Guizhou University, in 2003, and the Ph.D. degree in computer application technology from Harbin Engineering University, in 2012. He is currently a Professor and a M.S. Supervisor with the Anhui University of Science and Technology. He used to be a Visiting Scholar at the University of Arkansas for one year. His research interests include AI security, computer vision, and privacy preserving.

YU ZHANG received the master’s degree in Geological Engineering from Anhui University of Science and Technology, in 2005. She is currently a lecture and used to be a Visiting Scholar at the University of Central Florida for one year. Her research interests include deep learning, mining and visualization of media data applications and techniques.

* * *

VOLUME 4, 2016

11