# Knowledge-Enriched Distributional Model Inversion Attacks

Si Chen, Mostafa Kahla, Ruoxi Jia
Virginia Tech
Blacksburg, VA
{chensi, kahla, ruoxijia}@vt.edu

Guo-Jun Qi*
Seattle Research Center, Innopeak Technology
Bellevue, WA
guojun.qi@innopeaktech.com

## Abstract

*Model inversion (MI) attacks are aimed at reconstructing training data from model parameters. Such attacks have triggered increasing concerns about privacy, especially given a growing number of online model repositories. However, existing MI attacks against deep neural networks (DNNs) have large room for performance improvement. We present a novel inversion-specific GAN that can better distill knowledge useful for performing attacks on private models from public data. In particular, we train the discriminator to differentiate not only the real and fake samples but the soft-labels provided by the target model. Moreover, unlike previous work that directly searches for a single data point to represent a target class, we propose to model a private data distribution for each target class. Our experiments show that the combination of these techniques can significantly boost the success rate of the state-of-the-art MI attacks by* $150\%$*, and generalize better to a variety of datasets and models. Our code is available at* https://github.com/SCccc21/Knowledge-Enriched-DMI.

## 1. Introduction

Many attractive applications of machine learning (ML) techniques involve training models on sensitive and proprietary datasets. One major concern for these applications is that models could be subject to privacy attacks and reveal inappropriate details of the training data. One type of privacy attacks is MI attacks, aimed at recovering training data from the access to a model. The access could either be black-box or white-box. In the blackbox setting, the attacker can only make prediction queries to the model, while in the whitebox setting, the attacker has complete knowledge of the model. Given a growing number of online platforms where users can download entire models, such as Tensorflow Hub[1] and ModelDepot[2], whitebox MI attacks have posed an increasingly serious threat to privacy.

Effective MI attacks have been mostly demonstrated on simple models, such as linear models, and low-dimensional feature space [5, 4]. MI attacks are typically cast as an optimization problem that seeks for the most likely input examples corresponding to a target label under the private model. When the target model is a DNN, the underlying attack optimization problem becomes intractable and solving it via gradient methods in an unconstrained manner may easily end in a local minima. Previous MI attack models like [31] explore the idea of distilling a generic prior from potential public data via a GAN generator and using it to guide the inversion process. For instance, to attack a face recognition classifier trained on private face images, one can train a GAN with public face datasets to learn generic statistics of real face images and then solving the attack optimization over the latent space of the GAN rather than in an unconstrained ambient space.

However, there still exists a large room to improve the attack performance. For instance, the top-one identification accuracy of face images inverted from the state-of-the-art face recognition classifier is 45%. A natural question is: *Is the underperformance of MI attacks against DNNs because DNNs do not memorize much about private data or it is simply an artifact of imperfect attack algorithm design?* This paper shows that it is the latter.

We reveal a variety of drawbacks associated with the the current MI attacks against DNNs. Particularly, we notice that the previous state-of-the-art approach suffers from the two key limitations: 1) The information about private classifier is not sufficiently explored for distilling knowledge from public data. Previous works ignore the important role of the target classifier in adapting the knowledge distilled from the public data for training the MI attack model on the target classifier. Indeed, given a target classifier to attack, we can also use its output labels to distill which public data are more useful in inverting the target model to recover the private training examples of the given labels. 2) Prior works made a simplified one-to-one assumption in recovering a single example for a given label of the target model. How-

*Correspondence to G.-J. Qi, guojunq@gmail.com
[1] https://www.tensorflow.org/hub
[2] https://modeldepot.io/

ever, in real scenarios, inverting a model should naturally result in a distribution of training examples corresponding to the given label. This inspires us to recover a data distribution in the MI attack in line with such a many-to-one assumption.

To address the first limitation, we propose to tailor the training objective of the GAN to the inversion task. Specifically, for the discriminator, we propose to leverage the target model to label the public dataset and train the discriminator to differentiate not only the real and fake samples but also the labels. This new training scheme will force the generator to retain image statistics that are more relevant to infer the classes of the target model, which are likely to occur in the unknown private training data. To overcome the second limitation, we propose to explicitly parameterize the private data distribution and solve the attack optimization over the distributional parameters. Moreover, this will lead us to explore a distribution in which each point with large probability mass will achieve a good attack performance. We perform experiments on various datasets and network architectures and show that such a distributional MI attack by distilling public-domain knowledge tailored for private labels can significantly improve the previous state-of-the-art attack against DNNs, even when the public data have no overlap with the private labels of the target network.

The paper is organized as follows. We introduce related works in Section 2 and describe our proposed inversion-specific GAN and distributional recovery in Section 3. In Section 4, we assess the performance of the proposed method and show the extend application to a new attack setting: multi-target MI attacks. Finally, we conclude and discuss our key findings in Section 5.

## 2. Related work

The general goal of privacy attacks against ML models is to gain knowledge which is not intended to be shared, such as knowledge about the training data and information about the model. Attacks can be categorized into four types according to the specific goals: model extraction [19, 14, 22, 3], membership inference [25], property inference [1, 6, 18], and model inversion [5, 4, 31, 28]. Model extraction attacks try to create a substitute model that learns the same task as the target model while performing equally good or even better; and the other three focus on exposing secrets about training data. MI attacks, which are of particular interest, aims to recreate training data or sensitive attributes.

The first MI attack algorithm was proposed in [5], which follows the Maximum a Posterior (MAP) principle and constructs the input features that maximize the likelihood of observing a given model response and other possible auxiliary information. The authors applied the algorithm to attack a linear regression model that predicts medical dosage and showed that the algorithm can successfully invert genetic markers which are used as part of the input features.

Fredrikson *et al.* [4] applied the MAP attack idea to more complex models, including decision trees and shallow neural networks. Specifically, for neural networks with high-dimensional input features, the authors proposed to utilize gradient descent to solve the underlying attack optimization problem. Although the algorithm significantly outperforms random guessing when tested on some shallow networks and single-channel images, the reconstructions are blurry and can hardly reveal private information. Besides, the algorithm completely fails when tested on DNNs and three-channel images.

To improve the attack performance for DNNs with high-dimensional input, a two-pronged attack approach [31] was proposed which trains a GAN on public data (which could have no class intersection with private data and no labels), and then uses the GAN to search for the real examples that maximize the response to given classes. However, the resultant GAN fails to distill the private knowledge customized for the specific classes of interest in the target network, and the associated MI attack cannot recover the distribution of examples corresponding to those private classes.

The aforementioned works for attacking neural nets focused on the white-box setting and attacking a single model that is learned offline. Recent work has also looked into other attacker models. For instance, Yang *el al.* [28] studied the blackbox attack and proposed to train a separate model that swaps the input and output of the target model to perform MI attacks. Salem *et al.* [23] studied the blackbox MI attacks for online learning, where the attacker has access to the versions of the target model before and after an online update and the goal is to recover the training data used to perform the update.

Moreover, the algorithms of MI attacks resemble an orthogonal line of work on feature visualization [21, 29], which also attempts to reconstruct an image that maximally activates a target network. The proposed work differs from these existing works on feature visualization in that our algorithm customizes the public-to-private knowledge distillation to train the GAN and a novel formulation is presented for data distribution synthesis which results in more realistic image recovery.

## 3. The Proposed Approach

### 3.1. Overview of our attack

**Attack model** This paper focuses on the whitebox MI attack, in which the attacker has complete access to the target network $T$. The goal of the attacker is to discover *a representative input feature* $x$ associated with a specific label $y$. We will use face recognition as a running example for the target network. Face recognition classifiers label an image

containing a face with the label corresponding to the identity depicted in the image. The corresponding attack goal is to recover a representative face image for any given identity based on the target classifier parameters.

Existing MI attacks boil down to synthesizing the most likely input for the target network. Specifically, the following optimization problem is solved to synthesize the input for a given label $y$: $\max_x \log T_y(x)$, where $T_y(x)$ is the probability of label $y$ output by the model $T$ given the input $x$. When $T$ is a DNN and $x$ is high-dimensional (e.g., images), the corresponding optimization becomes nonconvex and performing gradient descent easily gets stuck in local minima, which might not be semantically meaningful at all. For instance, when the model input is an image, such local minima could be meaningless patterns of pixels.

The proposed proposed attack algorithm consists of two steps. The first step is to train a GAN having knowledge about the private classes of the target model from public data. Instead of training a generic GAN, we customize the training objective for both generator and discriminator so as to better distill the private-domain information about the target model from public data. In the second step, we make use of the generator learned in the first step to estimate the parameters of the private data distribution. The overall architecture of our method is shown in Figure 1.

### 3.2. Building an Inversion-Specific GAN

To distill the useful knowledge about the target model from public data, we propose to adopt a discriminator that is not only able to differentiate real data from the fake, but also to discriminate between the class labels under the target network.

Suppose that the target network classifies a sample into one of $K$ possible classes. Our discriminator $D$ is a $(K+1)$-classifier [24], where the first $K$ classes correspond to the labels of the target network and the $(K+1)$-th class represents fake samples. To train such a discriminator, we use the target network $T$ to generate a soft label $T(x)$ for each image from the public dataset.

Formally, the training loss for $D$ has two parts:

$$L_D = L_{\text{supervised}} + L_{\text{unsupervised}} \tag{1}$$

where

$$L_{\text{supervised}} = -\mathbb{E}_{x\sim p_{\text{data}}(x)} \sum_{k=1}^{K} T_k(x) \log p_{\text{disc}}(y=k \mid x) \tag{2}$$

and

$$L_{\text{unsupervised}} = -\{\mathbb{E}_{x\sim p_{\text{data}}(x)} \log D(x) + \tag{3}$$

$$\mathbb{E}_{z\sim \text{noise}} \log(1 - D(G(z)))\}. \tag{4}$$

Here $p_{\text{data}}$ is the distribution of public data, and $p_{\text{disc}}(y|x)$ is the probability that the discriminator predicts $x$ as class $y$. The random noise $z$ is sampled from $\mathcal{N}(0, I)$, and $T_k(x)$ is the $k$-th dimension of the soft label produced by the target network. The discriminator $D(x)$ outputs the probability of $x$ being a real sample, and therefore we have $D(x) \triangleq p_{\text{disc}}(y < K+1|x)$.

Intuitively, using the public data with soft-labels to train the discriminator encourages the generator to produce image statistics that help predict the output classes of the target model. Such image statistics are also likely to be present in the private training data. Hence, the proposed training process can potentially guide the generator to produce images that share more common characteristics with the private training data.

For training the generator, we adopt the following feature-matching loss [24] to align the generated images with the real counterparts based on the learned features $\mathbf{f}(x)$ encoded in an intermediate layer of the discriminator:

$$L_G = \|\mathbb{E}_{x\sim p_{\text{data}}} \mathbf{f}(x) - \mathbb{E}_{z\sim \text{noise}} \mathbf{f}(G(z))\|_2^2 + \lambda_h L_{\text{entropy}} \tag{5}$$

where $L_{\text{entropy}}$ is an entropy regularizer [7].

The intuition of the entropy regularization term is simple. Because the target network is trained on the private data, the private data should have high confidence when fed into the target network and in turn should get low prediction entropy. In order to encourage the data distribution learned from public data to mimic the private data, we explicitly constrain the entropy in the loss function so that the generated data will have low entropy under the target network.

### 3.3. Distributional Recovery

Given the GAN trained above on the public data under the guidance of the target network, the second step of the MI attack tries to find the private data which achieves the maximum likelihood under the target classification network while containing realistic images. While existing works focus on generating a representative image of a given identity, there ought to be a variety of training examples corresponding to one identity – indeed, the classifier is a many-to-one mapping. To this end, we propose to recover a data distribution instead of a single point to invert the target model for a given label $k$ of identity.

Specifically, given an identity label $k$, we model the private data distribution by $G(z')$, where $G$ is the generator trained in the first step and $z'$ is sampled from $p_{\text{gen}} = \mathcal{N}(\mu, \sigma^2)$ with two *learnable* parameters $\mu$ and $\sigma^2$. We then minimize the following objective function to generate the samples for the given class $k$ from the private classifier $T$ by estimating $\mu$ and $\sigma$:

$$L = L_{\text{prior}} + \lambda_i L_{\text{id}} \tag{6}$$

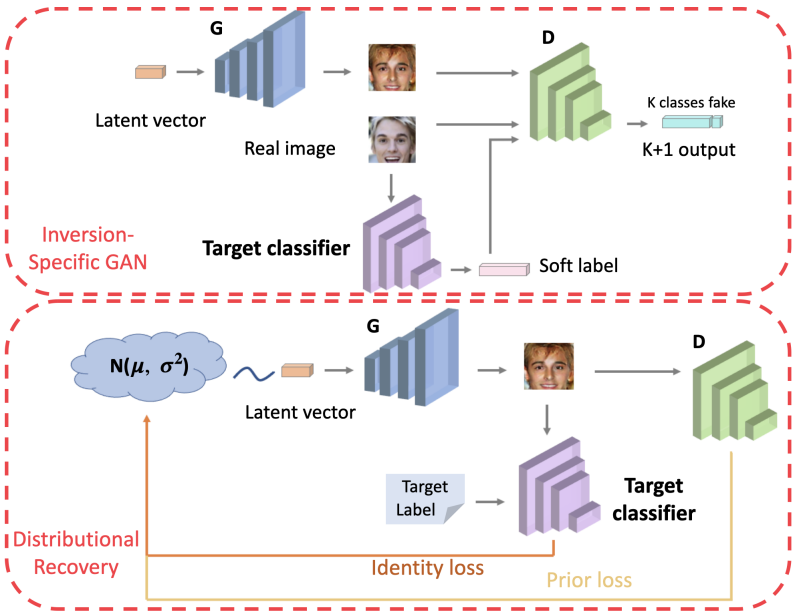


Figure 1. Overall architecture of the proposed attack algorithm. *Step 1.* Build an inversion-specific GAN to distill private information. *Step 2.* Recover the distribution of private domain. Note that both the generator and discriminator are fixed at Step 2.

where $\lambda_i$ is a positive balancing hyperparameter, and

$$L_{\text{prior}} = -\mathbb{E}_{z' \sim p_{\text{gen}}} \log D(G(z')) \tag{7}$$
$$L_{\text{id}} = -\mathbb{E}_{z' \sim p_{\text{gen}}} T_k(G(z')) \tag{8}$$

Here the prior loss $L_{\text{prior}}$ penalizes unrealistic images and the identity loss $L_{\text{id}}$ encourages the estimated private data distribution to have high likelihood of being assigned to the given target label $k$ under the targeted network $T$.

To estimate $\mu$ and $\sigma^2$ directly through the back-propagation, we adopt the reparameterization trick [13] to make $L_{\text{prior}}$ and $L_{\text{id}}$ differentiable:

$$z' = \sigma\epsilon + \mu, \epsilon \sim \mathcal{N}(0, I) \tag{9}$$

We can now form Monte Carlo estimates of expectations of $L_{\text{prior}}$ and $L_{\text{id}}$ as follows and optimize them with respect to $\sigma$ and $\mu$:

$$L_{\text{prior}} = -\frac{1}{L}\sum_{l=1}^{L} \log D(G(\sigma\epsilon_l + \mu)) \tag{10}$$
$$L_{\text{id}} = -\frac{1}{L}\sum_{l=1}^{L} \log T_k(G(\sigma\epsilon_l + \mu)) \tag{11}$$

where $\epsilon_l \sim \mathcal{N}(0, I)$ for $l = 1, \ldots, L$.

Once $\mu$ and $\sigma$ are estimated, the distribution of the learned training examples corresponding to the label $k$ is given implicitly by sampling from $G(z')$ with $z' \sim \mathcal{N}(\mu, \sigma^2)$. Figure 2 shows some examples obtained by sampling from $G(z')$. These examples show a variety of face

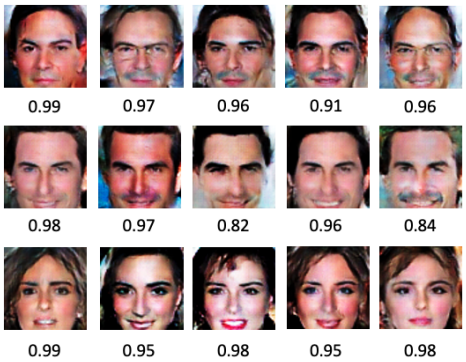


Figure 2. Examples of images obtained by inverting a target face recognition model. Each row corresponds to an identity. The numbers beneath each image show high softmax scores for the corresponding identity by the evaluation classifier, demonstrating these generated images successfully attack the target model by exposing its private information.

images are obtained for each identity by inverting a target face recognition model, containing variations in face poses, expressions, hairs and beards. This suggests that a natural many (faces)-to-one (identity) mapping is learned through a MI attack. We can also model the distribution by multi-variant Gaussian to have further improvement. And this will be left for future work.

## 4. Experiment

In this section, we will evaluate our proposed attack in terms of the performance to recover a representative in-

put from a target model. The baseline that we will compare against is the generative MI attack (GMI) proposed in [31], which achieved the state-of-the-art result for attacking DNNs.

### 4.1. Experimental setting

**Dataset.** We study attacks against models built for different prediction tasks, including face recognition, digit classification, object classification, and disease prediction. For face recognition, we use (1) the CelebFaces Attributes Dataset [17] (CelebA) containing 202,599 face images of 10,177 identities with coarse alignment, (2) Flickr-Faces-HQ (FFHQ) Dataset containing 70,000 high-quality images with considerable variation in terms of age, ethnicity and image background, and (3) FaceScrub consisting of 106,863 face images of male and female 530 celebrities, with about 200 images per person. We use aligned versions of above face datasets, and crop the images at the center and resize them to $64 \times 64$ so as to remove most background. For digit classification, we use the MNIST handwritten digit data [16]. For object classification, we adopt the CIFAR-10 dataset [15]. For disease prediction, we use the Chest X-ray Database [27] (ChestX-ray8).

**Models.** Following the settings in [31], we implement several different target networks with varied complexities. Some of the networks are adapted from existing ones by adjusting the number of outputs of their last fully connected layer to our tasks. For the face recognition task, we use three different network architectures: (1) VGG16 adapted from [26]; (2) ResNet-152 adapted from [9]; (3) face.evoLve adpated from [2]. For digit classification on MNIST, we use a network which consists of 3 convolutional layers and 2 pooling layers. For object classification, we use VGG16. For the disease prediction on ChestX-ray8, we use Resnet-18 adapted from [9].

**Attack Implementation.** We split each dataset into two disjoint parts: one part used as the private dataset to train the target network and the other as a public dataset. *The public data, throughout the experiments, do not have class intersection with the private training data of the target network*. Therefore, the public dataset in our experiment only helps the adversary to gain knowledge about features generic to all classes and does not provide information about private, class-specific features for training the target network. For CelebA, we use 30,027 images of 1000 identities as private set and randomly choose 30,000 images of other identities as public set to train GAN. For MNIST and CIFAR10, we use all of the images with label 0, 1, 2, 3, 4 as private set and rest images with label 5, 6, 7, 8, 9 as public set. For ChestX-ray8, we use 10,000 images with label "Atelectasis", "Cardiomegaly", "Effusion", "Infiltration", "Mass", "Nodule", "Pneumonia" as private set and 10,000 images belongs to other 7 classes as public set. We train the target networks using the SGD optimizer with the learning rate $10^{-2}$, batch size 64, momentum 0.9 and weight decay $10^{-4}$. For training GANs, we use the Adam optimizer with the learning rate 0.004, batch size 64, $\beta_1 = 0.5$ and $\beta_2 = 0.999$ as [12]. The weight for entropy regularization term is $\lambda_h = 1\text{e}{-4}$. For the step of distributional recovery, we set $\lambda_i = 100$; the distribution is initialized with $\mu = 0, \sigma = 1$ and optimized for 1500 iterations.

**Evaluation Protocol.** For our proposed attack, we draw 5 random samples of $\epsilon$ and generate corresponding images $G(\sigma\epsilon + \mu)$. For the baseline attack, we re-start the attack for 5 times with random initialization. To evaluate the reconstruction of a representative input, we compute the average of attack performance on the 5 reconstructed images.

**Evaluation Metrics.** Evaluating the MI attack performance requires gauging the amount of private information about a target label leaked through the synthesized images. We conduct both qualitative evaluation through visual inspection as well as quantitative evaluation. The quantitative metrics that we use to evaluate the attack performance largely follow the existing literature [31], including attack accuracy and K-nearest neighbor feature distance. They are generally aimed at measuring the semantic similarity between private data and reconstructions. In addition, we incorporate a metric for image quality, namely, Fréchet Inception Distance (FID) [10], as part of our evaluation. The metrics are expounded as follows.

- Attack Accuracy (Attack Acc). We build an *evaluation classifier* that predicts the identity based on the input reconstructed image. If the evaluation classifier achieves high accuracy, the reconstructed image is considered to expose private information about the target label. It is shown in [31] that the reconstructed images may overfit the target network; in other words, reconstructed images could be meaningless pixel patterns but achieve high prediction accuracy when evaluated with the target network. Hence, the evaluation classifier should be different from the target network. Moreover, the evaluation classifier should achieve high performance, because we are using it as a proxy for a human observer or an oracle to judge whether a reconstruction captures sensitive information. The attack accuracy is measured by the prediction accuracy of the evaluation classifier on reconstructed images. For all the face image datasets, we use the model in [2] as our evaluation classifier, which is pretrained on MS-Celeb-1M [8] and fine-tuned on the training set of the target networks. For MNIST, we train a new evaluation clas-

sifier which consists of 5 convolutional layers and 2 pooling layers on all of the 10 digits. For ChestX-ray8, the evaluation classifier is adapted from [26]. For CIFAR10, we use ResNet-18 adapted from [9].

- K-Nearest Neighbor Distance (KNN Dist). KNN Dist is the shortest feature distance from a reconstructed image to the real private training images for a given class. The feature distance is measured by the $l_2$ distance between two images when projected onto the feature space, i.e., the output of the penultimate layer of the evaluation classifier.

- FID. FID score measures feature distances between real and fake images, and lower FID values indicate better image quality and diversity. We found that reconstructed images which the evaluation classifier predicts into the target label tend to achieve lower FID scores. Hence, the FID score and attack accuracy are correlated with one another. To make FID a complementary metric to attack accuracy, we only calculate the FID score of those reconstructions which are successfully recognized as the target class by the evaluation classifier. The idea of this FID score is to measure how much more detailed information is leaked from a reconstruction that can successfully recover the semantics.

### 4.2. Result

**Comparison with previous state-of-the-art.** We compare our attack with the baseline for attacking various models built on the same dataset, namely, CelebA. The models include VGG16, ResNet152, and face.evolve, which have increased complexity. Among these models, face.evolve achieves state-of-the-art face recognition performance. The results for attacking these models are shown in Table 1, showing that our approach significantly improves the GMI on all the target models. Notably, our approach also enjoys lower performance variance across different target identities compared with the GMI.

The performance improvement achieved by our attack is further corroborated by Figure 3, which exhibits ground truth private images and corresponding reconstructions given by our attack and the GMI. We can see that our reconstructions can mostly better preserve the facial features of a given identity than the baseline. Since both our approach and the GMI are based on a GAN trained over public data, a natural question is whether these two approaches simply memorize the public data and output a public examples similar to the target identity? To answer this question, we also exhibit the nearest neighbors in the public dataset for each of the target images in Figure 3. We calculate the nearest neighbors based on the distance between deep feature representations extracted from the evaluation classifier in order to capture the perceptual similarity between two images [30]. Comparison between the nearest neighbors and our generated samples shows that both GMI and our approach do not simply "memorize" the similar images in the public domain; instead, they attempt to synthesize new images that expose sensitive attributes while remaining realistic.

Moreover, we examine the performance of the proposed attack for recovering some implicit attributes of the private images, such as gender, age, hair style, among others. Table 2 shows that our attack also outperforms GMI in terms of recovering the implicit attributes.

Table 3 compares the attack performance of our attack and the GMI on various datasets. We can see that our method outperforms the GMI by a large margin. One interesting finding is that, when attacking digit recognition model trained on MNIST, GMI generates images that can be successfully recognized as the target digits by the target classifier but cannot be predicted into the target digits by the evaluation classifier and the average attack accuracy is close to 0. As shown in Figure 4, when attacking digit "0," GMI tends to generate "6" because it only sees "6" in the public data. However, the generated samples can achieve high prediction accuracy under the target network, because it is trained to only predict 0-4, while having low prediction accuracy under the evaluation classifier which can predict all ten digits. In contrast, our attack can successfully reconstruct "0" even though it also only sees 5-9 in the public data. This demonstrates that our customized training of GAN can indeed help retain those features in the public data that are more likely to appear in private data.

**Cross-dataset experiment.** We study the effect of distribution shift between public and private data on the attack performance. We train our GAN on Flickr-Faces-HQ Dataset (FFHQ) [11] and FaceScrub [20] to attack the target network VGG16 trained on CelebA. The attack results are presented in Table 4, which shows that both GMI and our attack suffer from a performance drop while ours still outperforms GMI. We notice that the performance drop on FaceSrub is larger than that on FFHQ. One possible reason is that images in FaceScrub have much lower resolution ($64 \times 64$), and there are a number of images under poor lighting conditions or only showing partial faces. This performance drop could potentially be resolved by using a GAN combined with unsupervised domain adaptation techniques and we will leave the exploration of this line of work to future work.

**Ablation study.** We proposed a couple of ideas to improve the GMI attack in [31], including (1) soft-label discrimination (SD), which enables the discriminator to differentiate soft-labels produced by the target network, (2)

| | face.evolve | | IR152 | | VGG16 | |
|---|---|---|---|---|---|---|
| | GMI | Ours | GMI | Ours | GMI | Ours |
| **Attack Acc** ↑ | .31±.0039 | **.81±.0016** | .32±.0027 | **.81±.0015** | .21±.0020 | **.72±.0018** |
| **Top-5 Attack acc** ↑ | .53±.0015 | **.96±.0004** | .57±.0005 | **.96±.0001** | .43±.0014 | **.92±.0003** |
| **KNN Dist** ↓ | 1703.52 | **1358.23** | 1673.05 | **1324.72** | 1772.50 | **1380.22** |
| **FID** ↓ | 33.81 | **25.28** | 50.11 | **26.35** | 52.51 | **23.72** |

Table 1: Attack performance comparison on various models trained on CelebA. ↑ and ↓ respectively symbolize that higher and lower scores give better attack performance.

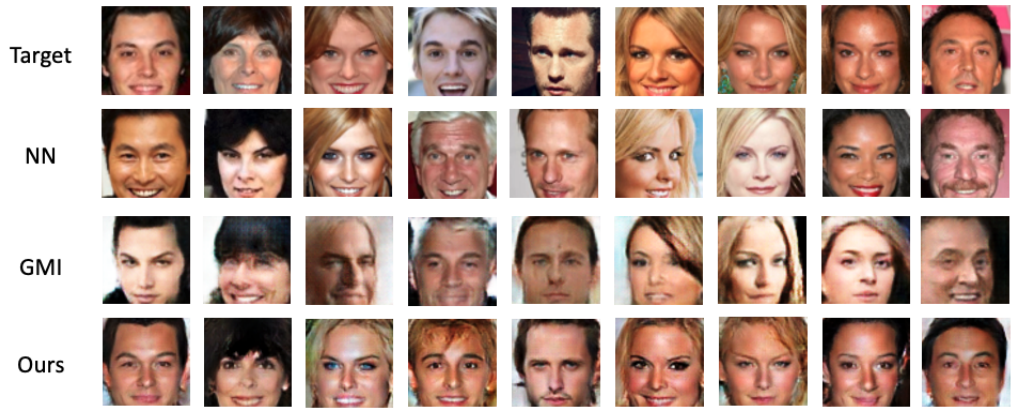


Figure 3. Qualitative comparison for attacking a face recognition model trained on CelebA. The first row shows ground truth images for target identities. The second row shows nearest neighbors of the target images from public domain. And the third and last rows demonstrate the reconstructions produced by the GMI attack and our attack, respectively.

| Attributes | Attack Acc ↑ | |
|---|---|---|
| | GMI | Ours |
| **Blond Hair** | 84 | **85** |
| **Bushy Eyebrows** | **85** | **85** |
| **Glasses** | 95 | **96** |
| **Male** | 86 | **94** |
| **Mustache** | 90 | **93** |
| **Young** | 72 | **82** |
| **5 o'clock shadow** | 83 | **87** |
| **Arched Eyebrows** | 65 | **70** |
| **Big Nose** | 73 | **78** |
| **Heavy Makeup** | 61 | **72** |
| **Narrow Eyes** | 78 | **82** |
| **No Beard** | 84 | **90** |
| **Wearing Lipstick** | 57 | **74** |

Table 2: Comparison of implicit attributes recovering between GMI and our proposed method. Attack accuracy is measured by an attributes classifier trained on CelebA.

entropy minimization (EM), which minimizes the prediction entropy of images produced by the generator, and (3) distributional recovery (DR), which explicitly models and estimates the private data distribution. Note that EM can only be combined with our SD. This is because a canonical


Figure 4. MNIST samples generated by GMI and our method when attacking digit "0".

discriminator only performs real vs. fake classification and minimizing the entropy of the prediction outputs in this case would not encourage the embedding to retain features that are more likely to appear in private data. We have shown that the combination of all these ideas can lead to significant attack performance improvement over the GMI. Here, we conduct an ablation study to investigate the improvement introduced by each individual idea as well as any reasonable combinations of these ideas. Table 5 presents the result of ablation study for attacking VGG16 trained on the CelebA dataset. We observe that both the attack accuracy and image quality get improved when we apply the idea of

| | CelebA | | MNIST | | ChestX-ray8 | | CIFAR10 | |
|---|---|---|---|---|---|---|---|---|
| | GMI | Ours | GMI | Ours | GMI | Ours | GMI | Ours |
| **Attack Acc** ↑ | .21±.0020 | **.72±.0018** | .08±.0120 | **.68±.0208** | .21±.0163 | **.47±.0155** | .56±.0264 | **.96±.0072** |
| **KNN Dist** ↓ | 1772.50 | **1380.22** | 126.61 | **72.54** | 360.32 | **220.30** | 139.09 | **123.07** |
| **FID** ↓ | 52.51 | **23.72** | 8.95 | **0.45** | 8.46 | **6.51** | 1.69 | **1.32** |

Table 3: Attack performance comparison on various datasets. ↑ and ↓ respectively symbolize that higher and lower scores give better attack performance.

| | FFHQ→CelebA | | FaceScrub→CelebA | |
|---|---|---|---|---|
| | GMI | Ours | GMI | Ours |
| **Acc** ↑ | .15±.0015 | **.36±.0015** | .03±.0004 | **.13±.0008** |
| **Acc5** ↑ | .35±.0017 | **.61±.0012** | .11±.0011 | **.30±.0015** |
| **KNN Dist** ↓ | 3014.45 | **2994.32** | 3003.90 | **2997.52** |
| **FID** ↓ | 69.12 | **36.02** | 112.83 | **60.05** |

Table 4: Attack performance comparison where there is large distributional shift between public and private data. $A \rightarrow B$ represents the setting when the target network is trained on dataset $B$ and the GAN is trained on dataset $A$ to distill a generic prior for reconstructions. ↑ and ↓ respectively symbolize that higher and lower scores give better attack performance.

SD or DR. Adding entropy minimization can further improve the performance. The combination of the three ideas leads to the largest improvement.

| | GMI | SD | SD+EM | DR | SD+DR | SD+EM+DR |
|---|---|---|---|---|---|---|
| **Acc** | .21±.0020 | .35±0042 | .43±.0035 | .47±.0022 | .62±.0028 | .72±.0018 |
| **Acc5** | .43±.0014 | .60±.0013 | .68±.0017 | .74±.0024 | .87±.0003 | .92±.0003 |
| **KNN Dist** | 1772.50 | 1653.53 | 1618.51 | 1562.48 | 1418.46 | 1380.22 |
| **FID** | 52.51 | 33.75 | 31.09 | 45.28 | 23.82 | 23.72 |

Table 5: Ablation study of ideas introduced in this paper, including soft-label discrimination (SD), entropy minimization (EM), and distributional recovery (DR).

| | F&I | | F&V | |
|---|---|---|---|---|
| | GMI | Ours | GMI | Ours |
| **Attack Acc** | .51±.0030 | **.90±.0009** | .51±.0048 | **.90±.0005** |
| **Top-5 Attack acc** | .78±.0025 | **.99±.0001** | .75±.0043 | **.98±.0002** |
| **KNN Dist** | 1527.94 | **1287.45** | 1528.32 | **1253.12** |
| **FID** | 54.89 | **29.37** | 54.76 | **28.66** |
| | I&V | | F&I&V | |
| | GMI | Ours | GMI | Ours |
| **Attack Acc** | .52±.0030 | **.92±.0008** | .67±.0030 | **.95±.0002** |
| **Top-5 Attack acc** | .79±.0023 | **.99±.0001** | .89±.0018 | **1±0** |
| **KNN Dist** | 1515.62 | **1251.02** | 1421.61 | **1216.96** |
| **FID** | 54.80 | **28.63** | 53.73 | **30.22** |

Table 6: Attack performance on CelebA under multi-target setting. F, I, and V refer to face.evolve, IR152, and VGG16 respectively.

**Extension to Multi-Target Model Inversion Attacks.** So far, existing MI attack methods mainly focus on attacking a single target model. It is interesting to study attack performance when multiple different models trained on the same private dataset are available. Will the attacker gain more information about this private dataset in this case? The proposed method can be easily extended to the multi-target MI attacks by combining the training losses over multiple target models. The details about the method are given in the supplementary material.

Table 6 shows the result of our method under the Multi-Target setting. We attack all possible combinations of the three target models used in the experiment shown in Table 1.

It is clear from Table 6 that the attack performance increases considerably under Multi-Target setting with both GMI and our approach. For example, When IR152 or face.evolve are jointly utilized with VGG16, their attack accuracy increased by 9% and 11% receptively over their accuracies under single-target setting, even though VGG16 yields a weaker attack accuracy. Moreover, increasing the number of target models to three models further improved the attack performance. By attacking multiple target models jointly, our approach achieves an attacking accuracy of over 0.9 in these experiments, which marks a significant milestone for multi-target MI attacks.

## 5. Conclusion

In this paper, we propose several techniques that can significantly improve whitebox MI attacks against DNNs. Specifically, we propose to customize the training of a GAN to better distill knowledge useful for performing inversion attacks from public data. Additionally, we propose to build an explicit parameteric model for the private data distribution and present methods to estimate its parameters. Our experiments show that the combination of the proposed techniques can lead to the state-of-the-art attack performance on various datasets, models, and even when the public data has a large distributional shift from private data. We also extend our work to a new attack setting where multiple models trained on the same private dataset are available. For future work, we will investigate the potential application of these techniques to improve the MI attack in the blackbox setting.

## References

[1] Giuseppe Ateniese, Luigi V Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali, and Giovanni Felici. Hacking smart machines with smarter ones: How to extract

meaningful data from machine learning classifiers. *International Journal of Security and Networks*, 10(3):137–150, 2015.

[2] Yu Cheng, Jian Zhao, Zhecan Wang, Yan Xu, Karlekar Jayashree, Shengmei Shen, and Jiashi Feng. Know you at one glance: A compact vector representation for low-shot learning. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1924–1932, 2017.

[3] Jacson Rodrigues Correia-Silva, Rodrigo F Berriel, Claudine Badue, Alberto F de Souza, and Thiago Oliveira-Santos. Copycat cnn: Stealing knowledge by persuading confession with random non-labeled data. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2018.

[4] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1322–1333, 2015.

[5] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, pages 17–32, 2014.

[6] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 619–633, 2018.

[7] Yves Grandvalet, Yoshua Bengio, et al. Semi-supervised learning by entropy minimization. In *CAP*, pages 281–296, 2005.

[8] Yandong Guo, Lei Zhang, Yuxiao Hu, Xiaodong He, and Jianfeng Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European conference on computer vision*, pages 87–102. Springer, 2016.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017.

[11] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[14] Kalpesh Krishna, Gaurav Singh Tomar, Ankur P Parikh, Nicolas Papernot, and Mohit Iyyer. Thieves on sesame street! model extraction of bert-based apis. *arXiv preprint arXiv:1910.12366*, 2019.

[15] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research).

[16] Yann Lecun, Leon Bottou, Y Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86:2278 – 2324, 12 1998.

[17] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[18] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 691–706. IEEE, 2019.

[19] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

[20] Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *2014 IEEE international conference on image processing (ICIP)*, pages 343–347. IEEE, 2014.

[21] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in neural information processing systems*, pages 3387–3395, 2016.

[22] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4954–4963, 2019.

[23] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. Updates-leak: Data set inference and reconstruction attacks in online learning. *arXiv preprint arXiv:1904.01067*, 2019.

[24] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

[25] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 3–18. IEEE, 2017.

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[27] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

[28] Ziqi Yang, Ee-Chien Chang, and Zhenkai Liang. Adversarial neural network inversion via auxiliary knowledge alignment. *arXiv preprint arXiv:1902.08552*, 2019.

[29] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

[30] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[31] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn 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*, pages 253–261, 2020.