Quantifying and Mitigating Privacy Risks of Contrastive Learning

Xinlei He and Yang Zhang
CISPA Helmholtz Center for Information Security

ABSTRACT

Data is the key factor to drive the development of machine learning (ML) during the past decade. However, high-quality data, in particular labeled data, is often hard and expensive to collect. To leverage large-scale unlabeled data, self-supervised learning, represented by contrastive learning, is introduced. The objective of contrastive learning is to map different views derived from a training sample (e.g., through data augmentation) closer in their representation space, while different views derived from different samples more distant. In this way, a contrastive model learns to generate informative representations for data samples, which are then used to perform downstream ML tasks. Recent research has shown that machine learning models are vulnerable to various privacy attacks. However, most of the current efforts concentrate on models trained with supervised learning. Meanwhile, data samples' informative representations learned with contrastive learning may cause severe privacy risks as well.

In this paper, we perform the first privacy analysis of contrastive learning through the lens of membership inference and attribute inference. Our experimental results show that contrastive models trained on image datasets are less vulnerable to membership inference attacks but more vulnerable to attribute inference attacks compared to supervised models. The former is due to the fact that contrastive models are less prone to overfitting, while the latter is caused by contrastive models' capability of representing data samples expressively. To remedy this situation, we propose the first privacy-preserving contrastive learning mechanism, *Talos*, relying on adversarial training. Empirical results show that *Talos* can successfully mitigate attribute inference risks for contrastive models while maintaining their membership privacy and model utility.[1]

CCS CONCEPTS

• **Security and privacy**; • **Computing methodologies** → **Machine learning**;

KEYWORDS

contrastive learning, membership inference attacks, attribute inference attacks, privacy-preserving machine learning

[1]Our code is available at https://github.com/xinleihe/ContrastiveLeaks.

CCS '21, November 15–19, 2021, Virtual Event, Republic of Korea

ACM ISBN 978-1-4503-8454-4/21/11…$15.00
https://doi.org/10.1145/3460120.3484571

ACM Reference Format:
Xinlei He and Yang Zhang. 2021. Quantifying and Mitigating Privacy Risks of Contrastive Learning. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21), November 15–19, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3460120.3484571

1 INTRODUCTION

Machine learning (ML) has progressed tremendously, and data is the key factor to drive such development. However, high-quality data, in particular labeled data, is often hard and expensive to collect as this relies on large-scale human annotation. Meanwhile, unlabeled data is being generated at every moment. To leverage unlabeled data for machine learning tasks, *self-supervised learning* has been introduced [34]. The goal of self-supervised learning is to derive labels from an unlabeled dataset and train an unsupervised task in a supervised manner. A trained self-supervised model serves as an encoder transforming data samples into their representations which are then used to perform supervised downstream ML tasks. One of the most prominent self-supervised learning paradigms is *contrastive learning* [9, 18, 20, 24, 29, 61, 67], with SimCLR [9] as its most representative framework [34].

Different from supervised learning which directly optimizes an ML model on a labeled training dataset, referred to as a supervised model, contrastive learning aims to train a contrastive model, which is able to generate expressive representations for data samples, and relies on such representations to perform downstream supervised ML tasks. The optimization objective for contrastive learning is to map different views derived from one training sample (e.g., through data augmentation) closer in the representation space while different views derived from different training samples more distant. By doing this, a contrastive model is capable of representing each sample in an informative way.

Recently, machine learning models have been demonstrated to be vulnerable to various privacy attacks against their training dataset [5, 7, 19, 22, 36, 49, 52, 55, 56]. The two most representative attacks in this domain are membership inference attack [49, 52] and attribute/property inference attack [36, 56]. The former aims to infer whether a data sample is part of a target ML model's training dataset. The latter leverages the overlearning property of a machine learning model to infer the sensitive attribute of a data sample. So far, most of the research on the privacy of machine learning concentrates on supervised models. Meanwhile, informative representations for data samples learned by contrastive models may cause severe privacy risks as well. To the best of our knowledge, this has been left largely unexplored.

Our Contributions. In this paper, we perform the first privacy quantification of contrastive learning, the most representative self-supervised learning paradigm. More specifically, we study the privacy risks of data samples in the contrastive learning setting, with a focus on SimCLR, through the lens of membership inference and attribute inference, and we concentrate on contrastive models trained on image datasets.

We adapt the existing attack methodologies for membership inference (neural network-based, metric-based, and label-only) and attribute inference against supervised models to contrastive models. Our empirical results show that contrastive models are less vulnerable to membership inference attacks than supervised models. For instance, considering the neural network-based attacks, we achieve 0.620 membership inference accuracy on a contrastive model trained on STL10 [11] with ResNet-50 [21], while the result is 0.810 on the corresponding supervised model. The reason behind this is contrastive models are less prone to overfitting.

On the other hand, we observe that contrastive models are more vulnerable to attribute inference attacks than supervised models. For instance, on the UTKFace [68] dataset with ResNet-18, we can achieve 0.701 attribute inference attack accuracy on the contrastive model while only 0.422 on the supervised model. This is due to the fact that the representations generated by a contrastive model contain rich and expressive information about their original data samples, which can be exploited for effective attribute inference.

To mitigate the attribute inference risks stemming from contrastive models, we propose the first privacy-preserving contrastive learning mechanism, namely *Talos*, relying on adversarial training. Concretely, *Talos* introduces an adversarial classifier into the original contrastive learning framework to censor the sensitive attributes learned by a contrastive model. Our evaluation reveals that *Talos* can successfully mitigate attribute inference risks for contrastive models while maintaining their membership privacy and model utility. Our code and models will be made publicly available.

In summary, we make the following contributions:

- We take the first step towards quantifying the privacy risks of contrastive learning.
- Our empirical evaluation shows that contrastive models trained on image datasets are less vulnerable to membership inference attacks but more prone to attribute inference attacks compared to supervised models.
- We propose the first privacy-preserving contrastive learning mechanism, which is able to protect the trained contrastive models from attribute inference attacks without jeopardizing their membership privacy and model utility.

2 PRELIMINARY

2.1 Supervised Learning

Supervised learning, represented by classification, is one of the most common and important ML applications. We first denote a set of data samples by X and a set of labels by Y. The objective of a supervised ML model $\mathcal{M}$ is to learn a mapping function from each data sample $x \in X$ to its label/class $y \in Y$. Formally, we have

$$\mathcal{M} : x \mapsto y \tag{1}$$

Given a sample x, its output from $\mathcal{M}$, denoted by $p = \mathcal{M}(x)$, is a vector that represents the probability distribution of the sample belonging to a certain class. In this paper, we refer to p as the prediction posteriors. To train an ML model, we need to define a loss function $\mathcal{L}(y, \mathcal{M}(x))$ which measures the distance between a sample's prediction posteriors and its label. The training process is then performed by minimizing the expectation of the loss function over a training dataset $\mathcal{D}^{train}$, i.e., the empirical loss. We formulate this as follow:

$$\arg\min_{\mathcal{M}} \frac{1}{|\mathcal{D}^{train}|} \sum_{(x,y)\in\mathcal{D}^{train}} \mathcal{L}(y, \mathcal{M}(x)) \tag{2}$$

Cross-entropy loss is one of the most common loss functions used for classification tasks, it is defined as the following.

$$\mathcal{L}_{CE}(y, p) = -\sum_{i=1}^{k} y^i \log p^i \tag{3}$$

Here, k is the total number of classes, y^i equals to 1 if the sample belongs to class i (otherwise 0), and p^i is the i-th element of the posteriors p. In this paper, we use cross-entropy as the loss function to train all the supervised models.

2.2 Contrastive Learning

Supervised learning is powerful, but its success heavily depends on the labeled training dataset. In the real world, high-quality labeled dataset is hard and expensive to obtain as it often relies on human annotation. For instance, the ILSVRC2011 dataset [47] contains more than 12 million labeled images that are all annotated by Amazon Mechanical Turk workers. Meanwhile, unlabeled data is being generated at every moment. To leverage large-scale unlabeled data, self-supervised learning is introduced.

The goal of self-supervised learning is to get labels from an unlabeled dataset for free so that one can train an unsupervised task on this unlabeled dataset in a supervised manner. Contrastive learning/loss [9, 18, 20, 24, 29, 61, 67] is one of the most successful and representative self-supervised learning paradigms in recent years and has received a lot of attention from both academia and industry. In general, contrastive learning aims to map a sample closer to its correlated views and more distant to other samples' correlated views. In this way, contrastive learning is able to learn an informative representation for each sample, which can then be leveraged to perform different downstream tasks. Contrastive learning relies on Noise Contrastive Estimation (NCE) [18] as its objective function, which can be formulated as:

$$\mathcal{L} = -\log\left(\frac{e^{sim(f(x),f(x^+))}}{e^{sim(f(x),f(x^+))} + e^{sim(f(x),f(x^-))}}\right) \tag{4}$$

where f is an encoder that maps a sample into its representation, x^+ is similar to x (referred to as a positive pair), x^- is dissimilar to x (referred to as a negative pair), and sim is a similarity function. The structure of the encoder and the similarity function can vary from different tasks. In this paper, we focus on one of the most popular contrastive learning frameworks [34], namely SimCLR [9]. This framework is assembled with the following components.

Data Augmentation. SimCLR first uses a data augmentation module to transform a given data sample x to its two augmented views,

denoted by $\tilde{x}_i$ and $\tilde{x}_j$, which can be considered as a positive pair for x. In our work, we follow the same data augmentation process used by SimCLR [9], i.e., first random cropping and flipping with resizing, second random color distortions, and third random Gaussian blur.

Base Encoder f. Base encoder f is used to extract representations from the augmented data samples. The base encoder can follow various neural network (NN) architectures. In this paper, we apply the widely used ResNet [21] (ResNet-18 and ResNet-50) and MobileNetV2 [50] to obtain the representation $h_i = f(\tilde{x}_i)$ for $\tilde{x}_i$.

Projection Head g. Projection head g is a simple neural network that maps the representations from the base encoder to another latent space to apply the contrastive loss. The goal of the projection head is to enhance the encoder's performance. Following Chen et al. [9], we implement it with a 2-layer MLP (multilayer perceptron) to obtain the output $z_i = g(h_i)$ for h_i.

Contrastive Loss Function. The contrastive loss function is defined to guide the model to learn the general representation from the data itself. Given a set of augmented samples $\{\tilde{x}_k\}$ including a positive pair $\tilde{x}_i$ and $\tilde{x}_j$, the contrastive loss maximizes the similarity between $\tilde{x}_i$ and $\tilde{x}_j$ and minimizes the similarity between $\tilde{x}_i$ ($\tilde{x}_j$) and other samples. For each mini-batch of N samples, we have $2N$ augmented samples. The loss function for a positive pair $\tilde{x}_i$ and $\tilde{x}_j$ can be formulated as:

$$\ell(i,j) = -\log \frac{e^{sim(z_i,z_j)/\tau}}{\sum_{k=1,k\neq i}^{2N} e^{sim(z_i,z_k)/\tau}} \tag{5}$$

where $sim(z_i, z_j) = {z_i}^\top z_j / \|z_i\| \|z_j\|$ represents the cosine similarity between z_i and z_j and τ is a temperature parameter. The final loss is calculated over all positive pairs in a mini-batch, which can be defined as the following.

$$\mathcal{L}_{Contrastive} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)] \tag{6}$$

Here, $2k-1$ and $2k$ are the indices for each positive pair.

Training classifiers with SimCLR can be partitioned into two phases. In the first phase, we train a base encoder as well as a projection head by the contrastive loss using an unlabeled dataset. After training, we discard the projection head and keep the base encoder only. In the second phase, to perform classification tasks, we freeze the parameters of the encoder, add a trainable linear layer at the end of the encoder, and fine-tune the linear layer with the cross-entropy loss (see Equation 3) on a labeled dataset. The linear layer serves as a classifier, with its input being the representations generated by the encoder. We refer to this linear layer as the *classification layer*. In the rest of the paper, we call a model trained with supervised learning as a *supervised model* and a model trained with contrastive learning as a *contrastive model*. Also, we consider contrastive models trained on image datasets, as most of the current development of contrastive learning focus on images.

Compared to supervised learning, contrastive learning can learn more informative representations for data samples. Previous work shows that supervised models are vulnerable to various privacy attacks [5, 7, 36, 49, 52, 56, 66]. However, to the best of our knowledge, privacy risks stemming from contrastive models have been left largely unexplored. In this work, we aim to fill this gap.

3 MEMBERSHIP INFERENCE ATTACK

We first quantify the privacy risks of contrastive models through the lens of membership inference. Note that our goal here is not to propose a novel membership inference attack, instead, we aim to quantify the membership privacy of contrastive models. Therefore, we follow existing attacks and their threat models [10, 33, 49, 52, 57].

3.1 Attack Definition and Threat Model

Membership inference attack is one of the most popular privacy attacks against ML models [7, 8, 10, 19, 28, 31, 33, 49, 52, 57]. The goal of membership inference is to determine whether a data sample x is part of the training dataset of a target model $\mathcal{T}$. We formally define a membership inference attack model $\mathcal{A}_{MemInf} : x, \mathcal{T} \mapsto \{member, non\text{-}member\}$. Here, the target model is the contrastive model introduced in Section 2. A successful membership inference attack can cause severe privacy risks. For instance, if a model is trained on data samples collected from people with certain sensitive information, then successfully inferring a sample from a person being a member of the model can directly reveal the person's sensitive information.

Following previous work [10, 33, 49, 52, 57], we assume that an adversary only has black-box access to the target model $\mathcal{T}$, i.e., they can only query $\mathcal{T}$ with their data samples and obtain the outputs. In addition, the adversary also has a shadow dataset $\mathcal{D}_{shadow}$, which comes from the same distribution as the target model's training dataset. The shadow dataset $\mathcal{D}_{shadow}$ is used to train a shadow model $\mathcal{S}$, the goal of which is to obtain the necessary information to perform the attack. We further assume that the shadow model shares the same architecture as the target model [52]. This is realistic as the adversary can use the same machine learning service as the target model owner to train their shadow model. Alternatively, the adversary can also learn the target model's architecture first by applying model extraction attacks [40, 41, 60, 63].

3.2 Methodology

We adapt the previous membership inference attacks, which are designed for supervised models, to contrastive models [10, 49, 52, 57]. Concretely, we consider three types of membership inference attacks, i.e., NN-based attacks [49, 52], metric-based attacks [57], and label-only attacks [10].

NN-based Attacks (Neural Network-based Attacks). In NN-based attacks, the adversary aims to train an attack model to differentiate members and non-members using the posteriors generated from the target model and their predicted labels. Given a shadow dataset $\mathcal{D}_{shadow}$, the adversary first splits it into two disjoint sets, namely shadow training dataset $\mathcal{D}^{train}_{shadow}$ and shadow testing dataset $\mathcal{D}^{test}_{shadow}$. $\mathcal{D}^{train}_{shadow}$ is used to train the shadow model $\mathcal{S}$, which mimics the behavior of the target model. This means the shadow model is trained to perform the same task as the target model. Then, the adversary uses $\mathcal{D}_{shadow}$ (including both $\mathcal{D}^{train}_{shadow}$ and $\mathcal{D}^{test}_{shadow}$) to query the shadow model $\mathcal{S}$ and obtains the corresponding posteriors and prediction labels. For each data sample in $\mathcal{D}_{shadow}$, the adversary ranks its posteriors in descending order and takes the largest two posteriors (classification tasks considered in this paper have at least two classes) as part of the input to the

attack model. The other part is an indicator representing whether the prediction is correct or not. Thus, the dimension of the input to $\mathcal{A}_{MemInf}$ is 3. If a sample belongs to $\mathcal{D}^{train}_{shadow}$, the adversary labels its corresponding input to the attack model as a member, otherwise as a non-member. Then, this obtained dataset is used to train the attack model, which is a binary machine learning classifier. To determine whether a target data sample x is used to train the target model, the adversary first queries the target model $\mathcal{T}$ with x and obtains the input to the attack model for this sample. Then, the adversary queries this input to the attack model and gets its membership prediction.

Metric-based Attacks. Song and Mittal [57] propose several metric-based attacks. Similar to NN-based attacks, metric-based attacks need to train shadow models. However, instead of training an attack model, metric-based attacks leverage a certain metric and a predefined threshold on that metric (calculated over the shadow model) to determine a sample's membership status. Song and Mittal [57] propose four metrics, i.e., prediction correctness (metric-corr), prediction confidence (metric-conf), prediction entropy (metric-ent), and modified prediction entropy (metric-ment).

Label-only Attacks. Label-only attacks [10] consider a more restrict scenario where the target model only exposes the predicted label instead of posteriors. Similar to previous attacks, this attack requires the adversary to train a shadow model. Label-only attacks focus more on the input samples instead of the model's outputs, relying on the adversarial example techniques. The key intuition is that the magnitude of perturbation to change the predicted label of member samples is larger than that of non-member samples. The adversary can exploit the magnitude of the perturbation to distinguish members and non-members.

3.3 Experimental Settings

Datasets. We utilize 8 different image datasets to conduct our experiments for membership inference.

- **CIFAR10 [1].** This dataset contains 60,000 images in 10 classes. Each class represents one object and has 6,000 images. The size of each image is 32×32.
- **CIFAR100 [1].** This dataset is similar to CIFAR10, except it has 100 classes, with each class containing 600 images. The size of each image is also 32×32.
- **STL10 [11].** This dataset is composed of 10 classes of images. Each class has 1,300 samples. The size of each image is 96×96. Besides the labeled image, STL10 also contains 100,000 unlabeled images, which we use for pretraining the encoder for the contrastive model (detailed later). These images are extracted from a broader distribution compared to those with labeled classes.
- **CelebA [35].** This dataset is composed of more than 200,000 celebrities' facial images. Note that in CelebA, we randomly select 60,000 images for our experiments. We set its target model's classification task as gender classification.
- **UTKFace [68].** This dataset consists of over 23,000 facial images labeled with gender, age, and race. We set its target model's classification task as gender classification as well.
- **Places365 [69].** This dataset is composed of more than 1.8 million images with 365 scene categories. We randomly select 100 scene categories and randomly select 400 images per category to form the **Places100** dataset. Besides, we randomly select 50 (20) scene categories and randomly select 800 (2,000) images per category to form the **Places50** (**Places20**) dataset. Each dataset contains 40,000 images in total. We follow Song and Shmatikov [56] and set its target model's classification task as predicting whether the scene is indoor or outdoor.

All the datasets are used to evaluate membership inference attacks, while UTKFace, Places100, Places50, and Places20 are also used to evaluate attribute inference attacks since they have extra labels that can be used as sensitive attributes (see Section 4.3). For all the datasets, we rescale their images to the size of 96×96. Note that we concentrate on image datasets as it is the most prominent domain for applying contrastive learning at the moment [9, 18, 20, 29, 61, 67]. We leave our investigation in other data domains as future work.

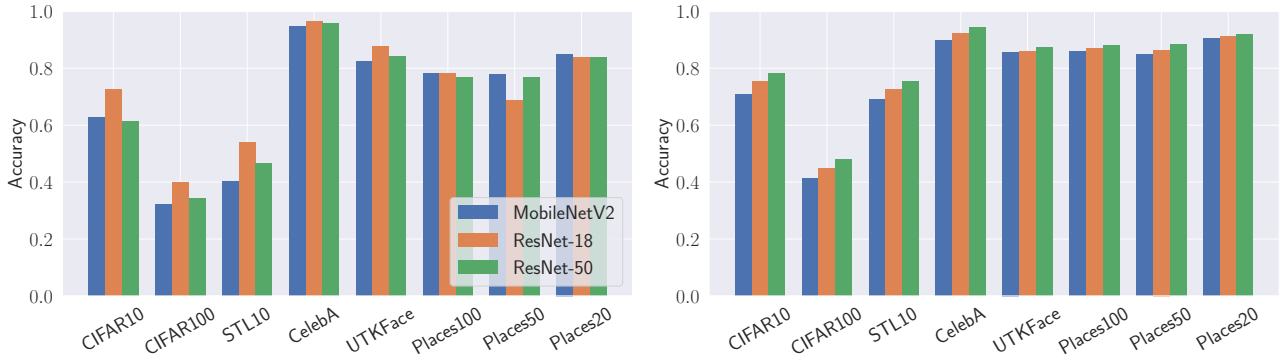
Datasets Configuration. For each dataset, we first split it into four equal parts, i.e., $\mathcal{D}^{train}_{target}$, $\mathcal{D}^{test}_{target}$, $\mathcal{D}^{train}_{shadow}$, and $\mathcal{D}^{test}_{shadow}$. $\mathcal{D}^{train}_{target}$ is used to train the target model $\mathcal{T}$, the samples of which are thus considered as members of the target model. We treat $\mathcal{D}^{test}_{target}$ as non-members of the target model $\mathcal{T}$. $\mathcal{D}^{train}_{shadow}$ is used to train the shadow model $\mathcal{S}$, and $\mathcal{D}^{train}_{shadow}$ and $\mathcal{D}^{test}_{shadow}$ are used to create the attack model $\mathcal{A}_{MemInf}$.

Metric. Since the attack model's training and testing datasets are both balanced with respect to membership distribution, we adopt accuracy as our evaluation metric following previous work [49, 52].

Attack Model. For NN-based attacks, the attack model is a 3-layer MLP, and the number of neurons for each hidden layer is set to 32. We use cross-entropy as the loss function and Adam as the optimizer with a learning rate of 0.05. The attack model is trained for 100 epochs. For metric-based attacks, we follow the implementation of Song et al. [57]. For label-only attacks, we leverage the implementation of ART [2].

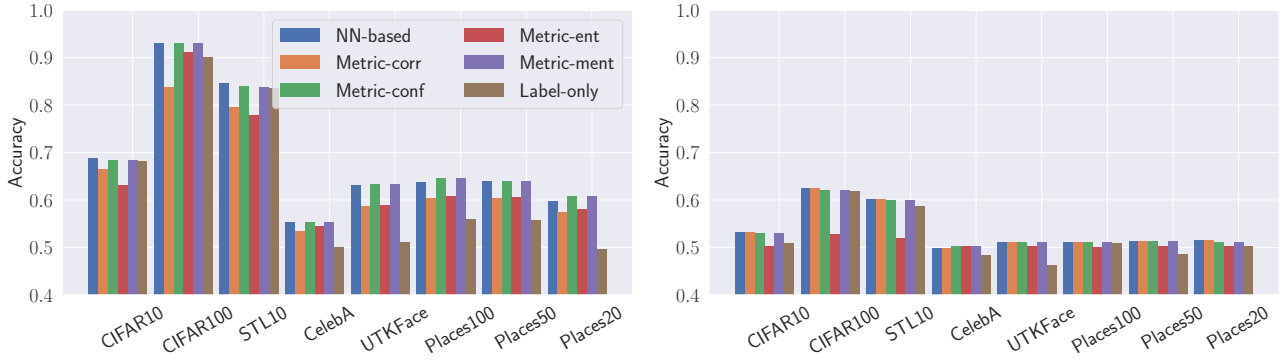
Target Model (Contrastive Model). We adopt three popular neural network architectures as the contrastive model's base encoder f in our experiments, including MobileNetV2 [50], ResNet-18 [21], and ResNet-50 [21]. Specifically, we discard the last classification layer of MobileNetV2, ResNet-18, and ResNet-50 and use the remaining parts as f. Then, a 2-layer MLP is added after f as the projection head g. For ResNet-18, the dimensions for the output of f, the first-layer of g, and the second-layer of g are set to 512, 512, and 256, respectively. For ResNet-50, the corresponding dimensions are 2,048, 256, and 256. For MobileNetV2, the corresponding dimensions are 1,280, 256, and 256.

After training the base encoder with the contrastive loss, we ignore the projection head g and add a new linear layer to the base encoder f as its classification layer. For all datasets, we first use the unlabeled dataset of STL10 to pretrain the base encoder f for 100 epochs. Then, we fine-tune the base encoder f with the corresponding training dataset (without label) for 100 epochs. In the end, we freeze the parameters of f and use the corresponding training dataset to only fine-tune the classification layer for 100



(a) Supervised Model **(b) Contrastive Model**

Figure 1: The performance of original classification tasks for both supervised models and contrastive models with MobileNetV2, ResNet-18, and ResNet-50 on 8 different datasets. The x-axis represents different datasets. The y-axis represents original classification tasks' accuracy.



(a) Supervised Model **(b) Contrastive Model**

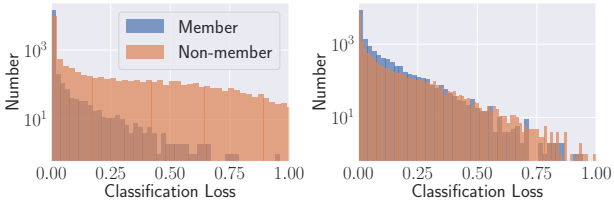
Figure 2: The performance of different membership inference attacks against both supervised models and contrastive models with MobileNetV2 on 8 different datasets. The x-axis represents different datasets. The y-axis represents membership inference attacks' accuracy.

epochs to establish the contrastive model. In all cases, Adam is utilized as the optimizer.

Baseline (Supervised Model). To fully understand the privacy leakage of contrastive models, we further use supervised models as the baseline. We train three models including MobileNetV2, ResNet-18, and ResNet-50 from scratch for all the datasets. The models are trained for 100 epochs. Cross-entropy is adopted as the loss function, and we again use Adam as the optimizer. Our code is currently implemented in Python 3.6 and PyTorch 1.6.0, and run on an NVIDIA DGX-A100 server with Ubuntu 18.04.

3.4 Results

We first show the performance of supervised models and contrastive models on their original classification tasks in Figure 1. We observe that contrastive models perform better than supervised models on most of the datasets. For instance, on STL10 with ResNet-18 as the base encoder, the contrastive model achieves 0.726 accuracy while the supervised model achieves 0.538 accuracy.



(a) Supervised Model **(b) Contrastive Model**

Figure 3: The distribution of loss with respect to original classification tasks for member and non-member samples for both the supervised model and the contrastive model with ResNet-18 on CIFAR10. The x-axis represents each sample's classification loss. The y-axis represents the number of member and non-member samples.

Regarding membership inference against supervised models and contrastive models, the results for MobileNetV2 are shown in Figure 2. We also summarize the results for ResNet-18 (Figure 15) and

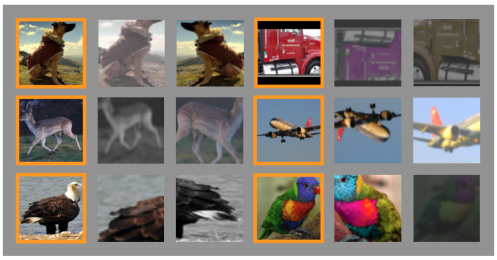


Figure 4: Randomly selected images from STL10 and their augmented views used during the process of contrastive learning. The first and fourth columns show the original images (bounded by orange boxes), and the rest columns show their augmented views.

ResNet-50 (Figure 16) in Appendix. In Figure 2, we see that all the supervised models have higher attack accuracy than the contrastive models. E.g., when the supervised model is MobileNetV2 trained on CIFAR100, the accuracy of NN-based attack is 0.931, while the accuracy for the corresponding contrastive model is only 0.625.

We observe that NN-based, metric-conf, and metric-ment attacks achieve the best performance in all cases. The reason metric-conf and metric-ment achieving better performance than metric-corr and metric-ent is that metric-conf and metric-ment consider both prediction correctness and confidence while metric-corr (metric-ent) only considers prediction correctness (confidence). Interestingly, for supervised models, metric-corr and metric-ent perform similarly, while for contrastive models, metric-ent is worse than metric-corr. This reason is that the posteriors generated by contrastive models are more smooth compared to supervised model, which makes it harder to distinguish members and non-members through the posterior entropy. Label-only attacks perform worse than NN-based attacks. This is expected since the adversary has less information in these cases. Note that label-only attacks do not perform well on binary classifiers, we will investigate the reason in the future.[2]

To further investigate why contrastive models are less vulnerable to membership inference, we analyze the loss distribution between members and non-members in both supervised models and contrastive models. Due to space limitations, we only show the results of ResNet-18 trained on the CIFAR10 dataset in Figure 3. A clear trend is that compared to the contrastive model, the supervised model has a larger divergence between the classification loss (cross-entropy) for members and non-members. Recall that contrastive learning uses two augmented views of each sample in each epoch to train its base encoder and the original sample to train its classification layer. This indicates that each sample is generalized to multiple views during the contrastive model training process. In this way, the contrastive model reduces its memorization of the original sample itself.

Interestingly, Song et al. [58] observe that defense mechanisms for mitigating adversarial examples [4, 6, 44, 59] increase the membership inference performance. This means such defense and contrastive learning have different effects on membership privacy. On the one hand, these defense mechanisms for adversarial examples use original samples and their visually imperceptible adversarial examples to train a model; in this way, the model learns to remember each original sample more accurately. On the other hand, the augmented samples in contrastive learning are very different from their original samples (see Figure 4 for some examples). Therefore, membership inference is less effective against contrastive models.

We notice that the attack performance varies on different models and different datasets. We relate this to the different overfitting levels. Similar to previous work [49, 52], we measure the overfitting level of a target model by calculating the difference between its training accuracy and testing accuracy. In Figure 5, we see that the overfitting level is highly correlated with the attack performance: if a model is more overfitted, it is more vulnerable to membership inference attacks. For instance, in Figure 5a, on CIFAR100, the contrastive model (upper right orange cross) has an overfitting level of 0.249, and the NN-based attack accuracy is 0.625, while the supervised model (upper right blue dot) has a larger overfitting level (0.678) and higher attack accuracy (0.931). Another observation is that compared to the supervised models, the overfitting levels of the contrastive models reside in a smaller range.

NN-based method as well as some of the metric-based ones (metric-ent and metric-ment) require the target model to provide posteriors to launch the attacks. We further investigate whether the number of posteriors provided by the target model can influence the attack performance. Concretely, we vary the number of posteriors from 2 to 100 on CIFAR100 for both supervised and contrastive models. Figure 6 shows that the number of posteriors does not have a strong influence on the attack performance. We further measure the influence of the number of epochs used for training each contrastive model's classification layer on the attack performance. Figure 7 shows that the attack accuracy is rather stable (the performance of metric-based attacks are summarized in Figure 17 in Appendix). These results show that contrastive models consistently reduce the membership threat.

In conclusion, contrastive models are less vulnerable to membership inference attacks compared to supervised models. The reason is that contrastive models are less overfitted to their training samples than supervised models due to the design of the contrastive learning paradigm.

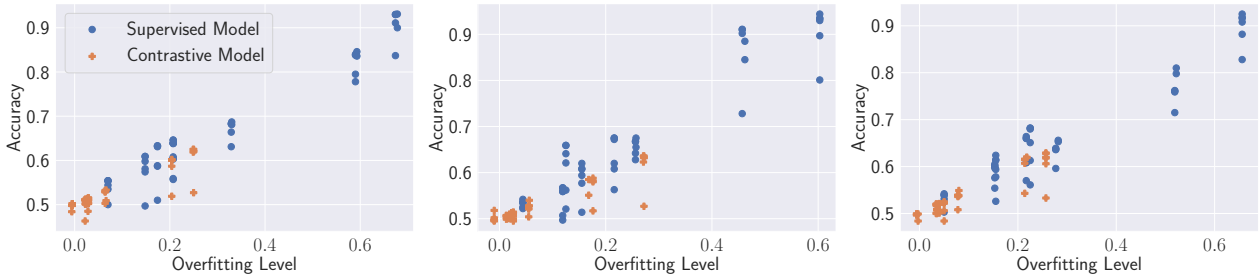
4 ATTRIBUTE INFERENCE ATTACK

In this section, we take a different angle to measure the privacy risks of contrastive learning using attribute inference attack [36, 56]. Similar to membership inference attacks, we use existing attribute inference attacks [36, 56] to measure the contrastive model's privacy risks instead of inventing new methods.

4.1 Attack Definition and Threat Model

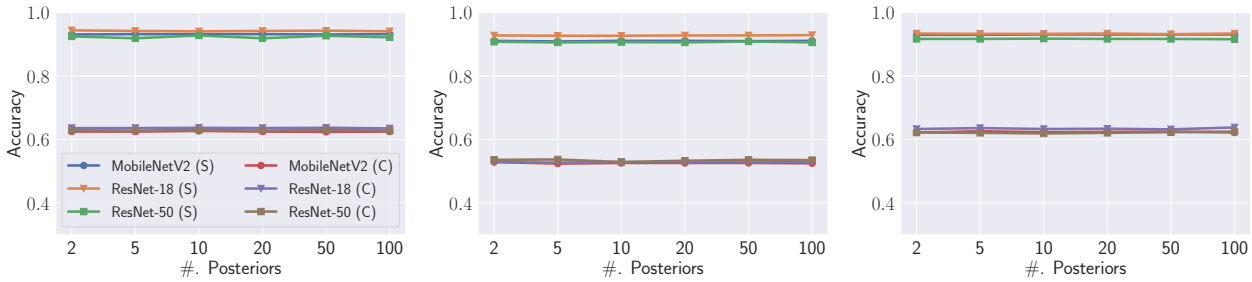
In attribute inference, the adversary's goal is to infer a specific sensitive attribute of a data sample from its representation generated by a target model. This sensitive attribute is not related to the target ML model's original classification task. For instance, a target model is designed to classify an individual's age from their social network posts, while attribute inference aims to infer their educational background.

[2] Choquette-Choo et al. [10] also only perform label-only membership inference attacks against datasets with more than two classes.



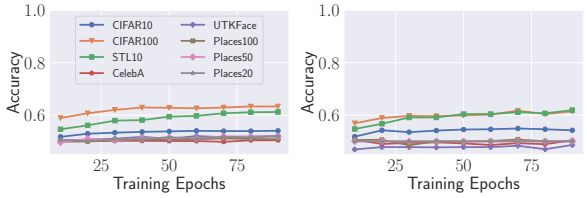
(a) MobileNetV2 (b) ResNet-18 (c) ResNet-50

Figure 5: The performance of membership inference attacks against both supervised models and contrastive models with MobileNetV2, ResNet-18, and ResNet-50 on 5 different datasets under different overfitting levels. The x-axis represents different overfitting levels. The y-axis represents membership inference attacks' accuracy.



(a) NN-based (b) Metric-ent (c) Metric-ment

Figure 6: The performance of NN-based, metric-ent, and metric-ment attacks against both supervised models and contrastive models with MobileNetV2, ResNet-18, and ResNet-50 on CIFAR100 under different numbers of posteriors given by the target models. (S) and (C) denotes the supervised and contrastive models, respectively. The x-axis represents different numbers of posteriors. The y-axis represents membership inference attacks' accuracy. Note that we do not report the performance of metric-corr, metric-conf, and label-only attacks since the number of posteriors does not affect their performance.



(a) NN-based (b) Label-only

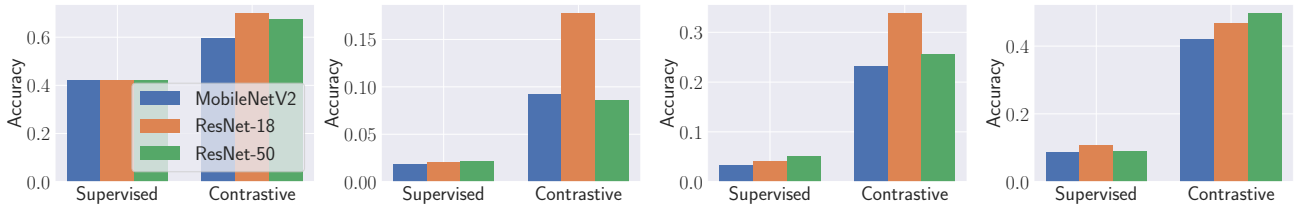
Figure 7: The performance of NN-based and label-only membership inference attacks against contrastive models with ResNet-50 on 8 different datasets under different numbers of epochs for classification layer training. The x-axis represents different numbers of epochs. The y-axis represents membership inference attacks' accuracy. Each line corresponds to a specific dataset.

Attribute inference attacks have been successfully performed on supervised models [36, 56]. The reason behind this is the intrinsic overlearning property of ML models. Overlearning means that an ML model trained for a certain task may also learn to represent other characteristics of data samples. Such representation capability, in some cases, can be exploited by an adversary to infer data samples' sensitive attributes.

Once a contrastive model is trained, it can generate a representation for each sample with its base encoder f. For a supervised model, we consider the whole model without the classification layer as its base encoder to generate a representation for each sample. Note that the base encoder of contrastive model and supervised model has the same architecture.

For attribute inference, given a data sample's representation from a target model, denoted by h, to conduct the attribute inference attack, the adversary trains an attack model $\mathcal{A}_{AttInf} : h \mapsto s$, where s represents the sensitive attribute.

We follow the same threat model as previous work [36, 56]: the adversary only has access to the target sample's embedding (representation), but not the target sample itself. The adversary is also assumed to have a set of samples' embeddings and their sensitive attributes; this dataset is termed as an auxiliary dataset $\mathcal{D}_{aux}$. As shown by previous work, attribute inference can be applied in both federated learning [36] and model partitioning [56] settings.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 8: The performance of attribute inference attacks against both supervised models and contrastive models with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents attribute inference attacks' accuracy.

4.2 Methodology

We generalize the methodology of attribute inference attacks against supervised models [36, 56] to contrastive models. The attack process can be partitioned into two stages, i.e., attack model training and attribute inference.

Attack Model Training. For each $(h, s) \in \mathcal{D}_{aux}$, the adversary takes the representation h as the input and the corresponding sensitive attribute s as the label to train the attack model.

Attribute Inference. To determine the sensitive attribute of a data sample's representation h, the adversary queries the attack model $\mathcal{A}_{AttInf}$ with h and obtains the result.

4.3 Experimental Setting

Datasets. We utilize UTKFace, Places100, Places50, and Places20 to evaluate attribute inference attacks as they contain extra attributes that can be considered as sensitive attributes for our experiments (see Section 3.3). In UTKFace, the target model's classification task is gender classification, and the sensitive attribute is race (Black, White, Asian, Indian, and Other). In Places100, Places50, and Places20, the target classification task is whether the scene is indoor or outdoor, and the sensitive attribmaxcute is scene categories. Similar to Song and Shmatikov [56], we take $\mathcal{D}_{target}^{train}$ to generate the auxiliary dataset and train the attack model, and take $\mathcal{D}_{target}^{test}$ to test the attack performance.

Metric. We adopt accuracy as the metric to evaluate attribute inference attacks following previous work [36, 56].

Models. All the target models' architectures are the same as those for membership inference attacks. For the attack model, we leverage a 3-layer MLP with the number of neurons in the hidden layer set to 128. We use cross-entropy as the loss function and SGD as the optimizer with a learning rate of 0.01. The attack model is trained for 100 epochs. The dimension of each sample's representation from the base encoder, i.e., the attack model's input, is 1,280 for MobileNetV2, 512 for ResNet-18, and 2,048 for ResNet-50.

4.4 Results

The performance of attribute inference attacks is depicted in Figure 8. First, we observe that, in general, attribute inference achieves effective performance except for the supervised model trained on UTKFace dataset (close to the prior sensitive attribute distribution in the attack training dataset as shown in Table 1). Second,

Table 1: The baseline accuracy (random guessing based on majority class labels) of attribute inference attack on different datasets.

Dataset	#. Class	Baseline Accuracy
UTKFace	5	0.421
Places100	100	0.012
Places50	50	0.023
Places20	20	0.053

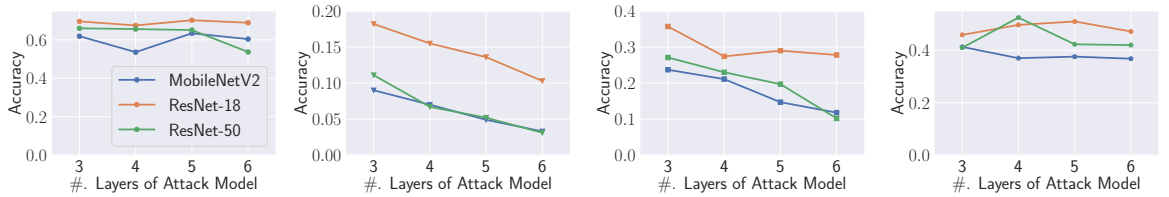
compared to the supervised models, the contrastive models are more vulnerable to attribute inference attacks. For instance, on the UTKFace dataset with ResNet-18, we can achieve an attack accuracy of 0.701 on the contrastive model while only 0.422 on the supervised model. To better understand this, we extract samples' representations (512-dimension) from ResNet-18 on UTKFace for both the supervised model and the contrastive model and project them into a 2-dimension space using t-Distributed Neighbor Embedding (t-SNE) [62]: Figure 9a shows the results for the supervised model on the original classification task, i.e., gender classification; Figure 9b shows the results for the supervised model on attribute inference, i.e., race. We see that in Figure 9a, male samples (blue) and female samples (orange) reside in completely different regions, which can be separated perfectly (the gender classification accuracy is 0.875 in Figure 1). However, for the sensitive attribute (Figure 9b), samples of different classes are clustered tightly, which increases the difficulty for attribute inference. Figure 9c and Figure 9d show the corresponding results for the contrastive model. We observe that different samples' representations on the contrastive model are less separable with respect to the original classification task compared to the supervised model (see Figure 9c and Figure 9a), but we can still successfully separate most of them correctly (the gender classification accuracy is 0.858 in Figure 1) since most of the male samples (blue) lie in the upper area while the female samples (orange) are in the lower area. On the other hand, for the sensitive attribute, compared to the supervised model (Figure 9b), representations generated by the contrastive model (Figure 9d) are more distinguishable. Our finding reveals that the representations generated by the contrastive model are more informative, which can be exploited not only for the original classification tasks but also for attribute inference.

(a) Original Task (S) (b) Sensitive Attribute (S) (c) Original Task (C) (d) Sensitive (C)

Figure 9: The representations for 200 randomly selected samples generated by both the supervised model and the contrastive model with ResNet-18 on UTKFace projected into a 2-dimension space using t-SNE. (S) and (C) denotes the supervised and contrastive models, respectively. Each point represents a sample.

(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 10: The performance of attribute inference attacks against contrastive models on 4 different datasets under different percentages of the attack training dataset. The x-axis represents different percentages of the attack training dataset. The y-axis represents attribute inference attacks' accuracy.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 11: The performance of attribute inference attacks against contrastive models on 4 different datasets under attack models with different layers. The x-axis represents attack models' layers. The y-axis represents attribute inference attacks' accuracy.

To study the effect of training dataset size on the attack model $\mathcal{A}_{AttInf}$, we randomly select from 10% to 90% of the training dataset to train the attack model and evaluate the performance using all the testing dataset; the results for contrastive models are summarized in Figure 10. By jointly considering Figure 8 and Figure 10, we can observe that, in most of the cases, even using 10% of the training dataset, the contrastive models are still more vulnerable to attribute inference attack than the supervised models when the attack model is trained with its *full* training dataset. On the other hand, the attack performance on supervised models is not significantly influenced by the training dataset size (see Figure 18). This further shows the privacy risks of contrastive learning.

Recall our attack model is a 3-layer MLP. We further investigate whether more complex attack models would improve the attack performance. To this end, we increase the attack model's layer from 3 to 6 and summarize the corresponding attack performance for contrastive and supervised models in Figure 11 and Figure 19 (in Appendix), respectively. The results show that 3-layer attack models can achieve the best performance in most of the cases. With more layers, the attack performance may degrade or keep stable, which indicates that even simple models are enough to launch effective attacks. This further shows that informative representations learned by contrastive models can be easily exploited by the adversary to infer samples' attributes.

We also observe that attribute inference attacks over contrastive models are more effective against smaller embedding size (see Figure 8 and Figure 10). For instance, ResNet-18 (512) leak more information than MobileNetV2 (1,280) and ResNet-50 (2,048). We conjecture that a larger embedding size represents each sample in a more complex space in the contrastive setting, which is harder for the attack model to decode. However, the effect of embedding size on attribute inference attacks against the supervised models is less pronounced (see Figure 8 and Figure 18 in the Appendix).

This further shows the difference between supervised models and contrastive models with respect to representing samples.

In conclusion, contrastive models are more vulnerable to attribute inference attacks compared to supervised models.

5 DEFENSE

So far, we have demonstrated that compared to supervised models, contrastive models are more vulnerable to attribute inference attacks (Section 4) but less vulnerable to membership inference attacks (Section 3). In this section, we propose the first privacy-preserving contrastive learning mechanism, namely *Talos*, which aims to reduce the risks of attribute inference for contrastive models while maintaining their membership privacy and model utility.

5.1 Methodology

Intuition. As shown in Section 4, the reason for a contrastive model to be vulnerable to attribute inference attacks is that the model's base encoder f learns informative representations for data samples, which can be exploited by an adversary. To mitigate such a threat, we aim for a new training paradigm for contrastive learning which can eliminate data samples' sensitive attributes from their representations. Meanwhile, the base encoder of the contrastive model still needs to represent data samples expressively for preserving model utility. These two objectives are in conflict, and our defense mechanism should consider both simultaneously.

Methodology. Our defense mechanism, namely *Talos*, can be modeled as a mini-max game, and we rely on adversarial training [12–14, 17, 65] to realize it. Similar to the original contrastive model, *Talos* also leverages a base encoder and a projection head to learn informative representations for data samples. Besides, *Talos* introduces an adversarial classifier C, which is used to censor sensitive attributes from data samples' representations.

The adversarial classifier of *Talos* is essentially designed for attribute inference. Similar to the original contrastive learning process (see Section 2), *Talos* is trained with mini-batches. Given a mini-batch of $2N$ augmented data samples (generated from N original samples), we define the loss of the adversarial classifier C as follows.

$$\mathcal{L}_C = \frac{1}{2N}\sum_{k=1}^{N}[\mathcal{L}_{CE}(s_k, C(f(\tilde{x}_{2k-1}))) + \mathcal{L}_{CE}(s_k, C(f(\tilde{x}_{2k})))] \quad (7)$$

where $\tilde{x}_{2k-1}$ and $\tilde{x}_{2k}$ are the two augmented samples of an original sample x_k, s_k represents x_k's sensitive attribute, f is the base encoder, and $\mathcal{L}_{CE}$ is the cross-entropy loss (Equation 3). We consider $\tilde{x}_{2k-1}$ and $\tilde{x}_{2k}$ sharing the same sensitive attribute as x_k. Note that we take the output of the base encoder instead of the projection head as the input to the adversarial classifier. Since the projection head is discarded after the first phase of training the contrastive model, directly optimizing the base encoder with the adversarial classifier loss would maintain the effect of adversarial training.

Talos also adopts the original contrastive loss $\mathcal{L}_{Contrastive}$ (Equation 6). By jointly considering the adversarial classifier loss and the contrastive loss, *Talos*'s loss function is defined as follows:

$$\mathcal{L}_{Talos} = \mathcal{L}_{Contrastive} - \lambda\mathcal{L}_C \quad (8)$$

where λ is the *adversarial factor* to balance the two losses. We refer to a model trained with *Talos* as a *Talos* model.

Algorithm 1: The training process of *Talos*.

1 **Input**: Target training dataset $\mathcal{D}^{train}_{target}$ with sensitive attribute s, base encoder f, projection head g, adversarial classifier C, and adversarial factor λ.
2 Initialize f, g, and C's parameters.
3 **for** ***each epoch*** **do**
4 **for** ***each mini-batch*** **do**
5 Sample a mini-batch with N training data samples and its corresponding sensitive attributes $\{(x_1, s_1), (x_2, s_2), ..., (x_N, s_N)\}$ from $\mathcal{D}^{train}_{target}$
6 Generate augmented data samples: $\{(\tilde{x}_1, s_1), (\tilde{x}_2, s_1), ..., (\tilde{x}_{2N}, s_N)\}$, where $\tilde{x}_{2k-1}$ and $\tilde{x}_{2k}$ are the two augmented views of x_k
7 Feed augmented data samples into the base encoder f and the projection head g to calculate the contrastive loss: $\mathcal{L}_{Contrastive} = \frac{1}{2N}\sum_{k=1}^{N}[\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
8 Feed the representations generated by the base encoder f into the adversarial classifier C to calculate the adversarial classifier loss: $\mathcal{L}_C = \frac{1}{2N}\sum_{k=1}^{N}[\mathcal{L}_{CE}(s_k, C(f(\tilde{x}_{2k-1}))) + \mathcal{L}_{CE}(s_k, C(f(\tilde{x}_{2k})))]$
9 **if** ***epoch*** mod $2 \neq 0$ **then**
10 Optimize adversarial classifier C's parameters with the adversarial classifier loss: $\mathcal{L}_C$
11 **else**
12 Optimize projection head g's parameters with the contrastive loss: $\mathcal{L}_{Contrastive}$
13 Optimize base encoder f's parameters with adversarial training loss: $\mathcal{L}_{Talos} = \mathcal{L}_{Contrastive} - \lambda\mathcal{L}_C$
14 **end**
15 **end**
16 **end**
17 **Return**: Base encoder f

Algorithm 1 presents the training process of *Talos*. In each mini-batch, given N training samples, we first generate $2N$ augmented views (Line 6) and feed them into the base encoder. The generated representations are then fed into the projection head (Line 7) and the adversarial classifier (Line 8) simultaneously. Note that the adversarial classifier and contrastive model are updated alternately by epoch. We First optimize the adversarial classifier with the cross-entropy loss (Line 10). Then we optimize the projection head with the contrastive loss (Line 12) and the base encoder with the loss function of *Talos*, i.e., Equation 8 (Line 13).

To implement this in practice, we utilize the gradient reversal layer (GRL) proposed by Ganin et al. [15]. GRL is a layer that can be added between the base encoder f and the adversarial classifier C. In the forward propagation, GRL acts as an identity transform that simply copies the input as the output. During the backpropagation, GRL takes the gradients passed through it from the adversarial classifier C, multiplies the gradients by $-\lambda$, and passes them to the base encoder f. Such operation lets the base encoder receive the

opposite direction of gradients from the adversarial classifier. In this way, the base encoder f is able to learn informative representations for samples while censoring their sensitive attributes.

Note that our adversarial training is performed only on the process of training the base encoder f. The training for the classification layer of the contrastive model remains unchanged. As we show in Section 3, the classification layer generalizes well on the contrastive models, i.e., less overfitting. Therefore, models trained by *Talos* should be robust against membership inference attacks as well. Our evaluation shows that this is indeed the case (see Figure 13).

Adaptive Attacks. An adversary needs to establish a shadow model to mount membership inference attacks. To evaluate membership privacy risks of *Talos*, we consider an adaptive (and stronger) adversary [28]. Concretely, we assume that the adversary knows the training details of *Talos* and trains their shadow model in the same way. For attribute inference, the attack model is trained on embeddings generated by *Talos*, thus, our attribute inference attack considered in the evaluation of *Talos* is also adaptive.

5.2 Experimental Setting

We follow the same experimental setting, including datasets, metrics, target models, and attack models (both attribute inference and membership inference), as those in Section 3.3 and Section 4.3. As mentioned before, both membership inference and attribute inference attacks are performed in an adaptive way. Regarding the adversarial classifier of *Talos*, we leverage a 3-layer MLP with 64 neurons in the hidden layer, which is smaller than the attribute inference attack model.

Baseline. We consider three state-of-the-art defenses, one for membership inference (*MemGuard* [28]) and two for attribute inference (*Olympus* [46] and *AttriGuard* [27]) as the baseline models. *MemGuard*, *Olympus*, and *AttriGuard* are originally designed for supervised models, here, we adapt them to contrastive models. Since the input to the attribute inference attack is each sample's representation, we further consider a sample's representation as the input to *Olympus* and *AttriGuard*.

MemGuard is a two-phase defense for membership inference. In phase I, the defender generates a noise vector to perturb the posteriors of a target sample, so that the adversary's membership classifier is likely to give a random guess for the perturbed posteriors. In phase II, the defender adds the noise vector to the posteriors with certain probability.

Olympus, designed for attribute inference, has three basic components: an autoencoder to transfer the original representation into the perturbed one, a classifier to perform the original task over the perturbed representation, and an adversarial classifier to infer the sensitive attribute from the perturbed representation. *Olympus* optimizes the three components using adversarial training to preserve the model utility while protecting samples' sensitive attributes. To perform *Olympus* on contrastive models, we first train a base encoder following the original contrastive learning process. Then, we add an autoencoder between the base encoder and the classification layer, and fine-tune the whole model using the original training samples with *Olympus*'s losses.

AttriGuard is a two-phase defense for attribute inference. In phase I, for each representation, the defender generates an adversarial example for each possible value of the sensitive attribute by adapting the existing evasion attack techniques. In phase II, the defender samples one sensitive attribute value based on a probability distribution and selects the corresponding adversarial example found in phase I as the new representation.

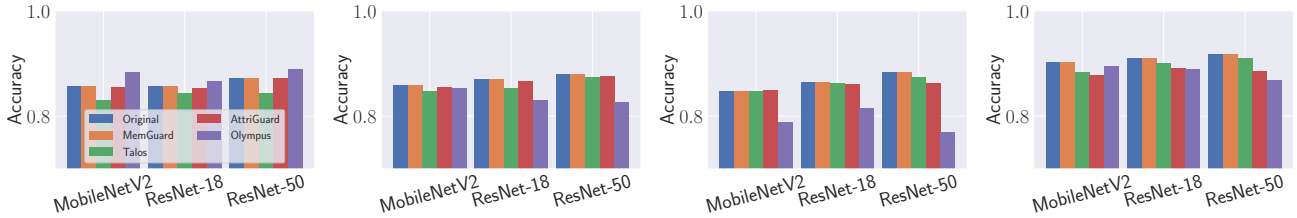
The adversarial classifier used in *AttriGuard* and *Olympus* shares the same architecutre as the one in *Talos*. For *MemGuard*, we follow Jia et al. [28] to generate the noise in Phase I. For the autoencoder of *Olympus*, we set its encoder (decoder) as a 2-layer MLP with 256 and 128 (128 and 256) neurons in the hidden layers. For *AttriGuard*, we leverage the C&W attack [6] with the L_{inf} norm in phase I.

5.3 Results

We compare the performance of the original classification tasks, NN-based membership inference attacks, and attribute inference attacks for the original contrastive model and the models defended by *Talos*, *MemGuard*, *Olympus*, and *AttriGuard*. The results are depicted in Figure 12, Figure 13, and Figure 14, respectively. Note that we also perform metric-based and label-only membership inference attacks and the results are summarized in Figure 20, Figure 21, Figure 22, Figure 23, and Figure 24 in Appendix.

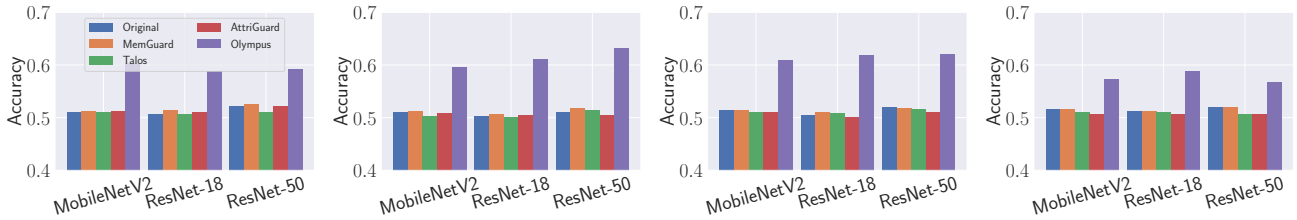
In Figure 14, we find that *Talos* indeed reduces the attribute inference accuracy compared to the original contrastive learning. For instance, the attribute inference accuracy is 0.701 on the original contrastive model with ResNet-18 on the UTKFace dataset, while only 0.602 on the *Talos* model. Meanwhile, the testing accuracy of the original classification task for the *Talos* model is also preserved (Figure 12).

For different defense mechanisms, we find that *Olympus* reduces attribute inference attacks the most (see Figure 14). However, it jeopardizes the membership privacy to a large extent (see Figure 13). For instance, the membership inference accuracy of the *Talos* model (ResNet-50) on Place100 is 0.513 while the corresponding *Olympus* model's value is 0.631. The reason is that *Olympus*'s training process utilizes the original training samples to fine-tune the whole model, which leads to the model memorizing these samples with the model's full capacity. On the other hand, as mentioned in Section 5.1, *Talos* is only performed on the training process of the base encoder f which considers each sample's augmented views. The original samples are only used to fine-tune the final classification layer, the same as training a normal contrastive model. In other words, the *Talos* model memorizes its training samples with only its one-layer capacity. Therefore, *Talos* models are less prone to membership inference. In addition, *Olympus* jeopardizes the target model's utility in multiple cases (see Figure 12b, Figure 12c, and Figure 12d), the reason again lies in the training process of *Olympus*. More specifically, *Olympus* needs to fine-tune the whole model in a supervised way, this reduces the effect of contrastive learning in the final model. Meanwhile, *Talos* preserves the contrastive learning process to a large extent as its adversarial loss is applied together with the contrastive loss during the training of the base encoder. Since membership privacy, attribute privacy, and model utility are equally important, we believe *Talos* is a better choice than *Olympus*.



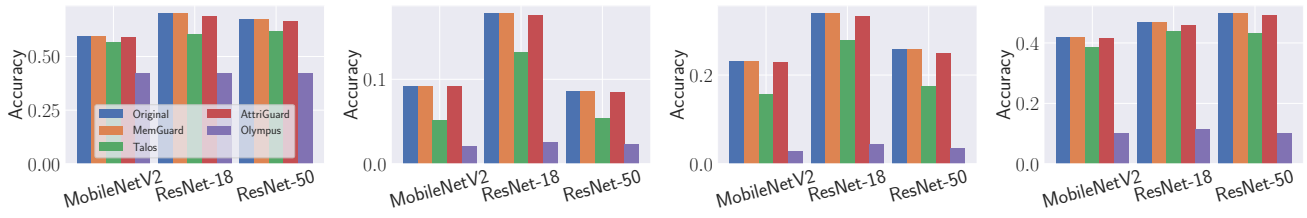
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 12: The performance of original classification tasks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of original classification tasks.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 13: The performance of NN-based membership inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of NN-based membership inference attacks.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 14: The performance of attribute inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of attribute inference attacks.

We also find that *Talos*, *MemGuard*, and *AttriGuard* models can achieve similar utility as the original contrastive models (see Figure 12). However, *Talos* can mitigate attribute inference attacks to a larger extent than *AttriGuard* and *MemGuard* (see Figure 14). For instance, the attribute inference accuracy is only 0.132 on the *Talos* model with ResNet-18 on the Places100 dataset, while 0.176 and 0.178 on the *AttriGuard* and *MemGuard* models. Also, as the contrastive learning procedure is preserved for *Talos*, *AttriGuard*, and *MemGuard*, we observe that all these defenses are robust against membership inference attacks (see Figure 13).

We also investigate the effect of the adversarial factor λ on the performance of original classification tasks, membership inference attacks, and attribute inference attacks. The results are summarized in Figure 25, Figure 26, and Figure 27. First of all, we observe that the performance of original classification tasks (Figure 25) and membership inference attacks (Figure 26) are relatively stable with respect to different adversarial factors. However, for different datasets or different model architectures, the best λ to defense attributes inference attack may vary (Figure 27). In general, we notice that setting λ to 2 or 3 can achieve nearly the best defense performance on most datasets and model architectures. To perform *Talos* in practice, we believe the model owner needs to tune the λ on their validation dataset. During the process, concentrating more on model utility or defense effectiveness depends on the ML model owner's purpose.

In conclusion, *Talos* can successfully defend attribute inference attacks for contrastive models without jeopardizing their membership privacy and model utility.

6 RELATED WORK

Contrastive Learning. Contrastive learning is one of the most popular self-supervised learning paradigms [9, 18, 20, 29, 61, 67]. Oord et al. [61] propose contrastive predictive coding, which leverages autoregressive models to predict future observations for data samples. Wu et al. [64] utilize a memory bank to save instance representation and k-nearest neighbors to conduct prediction. He et al. [20] introduce MoCo, which relies on momentum to update the key encoder with the query encoder to maintain consistency. Chen et al. [9] propose SimCLR, which leverages data augmentation and the projection head to enhance the performance of contrastive models. SimCLR is the most prominent contrastive learning paradigm at the moment [34], thus we concentrate on it in this paper.

Membership Inference Attack. In membership inference, the adversary's goal is to infer whether a given data sample is used to train a target model. Right now, membership inference is one of the major means to measure privacy risks of machine learning models [19, 23, 31, 38, 49, 52, 58, 66]. Shokri et al. [52] propose the first membership inference attack in the black-box setting. Specifically, they rely on training multiple shadow models to mimic the behavior of a target model to derive the data for training their attack models. Salem et al. [49] further relax the assumptions made by Shokri et al. [52] and propose three novel attacks. Later, Nasr et al. [38] conduct a comprehensive analysis of membership privacy under both black-box and white-box settings for centralized as well as federated learning scenarios. Song et al. [58] study the synergy between adversarial example and membership inference and show that membership privacy risks increase when a model owner applies measures to defend against adversarial example attacks. To mitigate membership inference, many defense mechanisms have been proposed [28, 37, 49]. Nasr et al. [37] introduce an adversarial regularization term into a target model's loss function. Salem et al. [49] propose to use dropout and model stacking to reduce model overfitting, the main reason behind the success of membership inference. Jia et al. [28] rely on adversarial examples to craft noise to add to a target sample's posteriors. Also, deferentially private methods [39, 45] are introduced to mitigate membership inference.

Attribute Inference Attack. Another major type of privacy attack against ML models is attribute inference. Here, an adversary aims to infer a specific sensitive attribute of a data sample from its representation generated by a target model [36, 56]. Melis et al. [36] propose the first attribute inference attack against machine learning, in particular federated learning. Song and Shmatikov [56] later show that attribute inference attacks are also effective against another training paradigm, namely model partitioning. They further demonstrate that the success of attribute inference is due to the overlearning behavior of ML models. More recently, Song and Raghunathan [53] demonstrate that language models are also vulnerable to attribute inference.

Other Attacks Against Machine Learning Models. Besides membership inference and attribute inference, there exist a plethora of other attacks against ML models [3, 5, 22, 26, 32, 42, 43, 48, 51, 54]. One major attack is adversarial example [4, 6, 44, 59], where an adversary aims to add imperceptible noises to data samples to evade a target ML model. Another representative attack in this domain is model extraction, the goal of which is to learn a target model's parameters [25, 30, 41, 60] or hyperparameters [40, 63].

7 DISCUSSION

Other Types of Datasets. In this paper, we only focus on image datasets, as most of the current efforts on contrastive learning concentrate on the image domain. For other types of datasets like texts or graphs, the main challenge is to define a suitable augmentation method for the input sample. There indeed exist some preliminary works of contrastive learning over texts or graphs [16, 67]. However, the effectiveness of these methods still needs to be further evaluated. We believe it is straightforward to extend our analysis to contrastive models trained on other types of data.

Novel Membership Inference Attacks Against Contrastive Models. Traditional membership inference attacks use the original data samples to query the model and get the corresponding posteriors to launch the attacks. However, such attacks is less effective on contrastive models as shown in our paper. Since the contrastive model is trained with some augmented views of each data sample, the model itself may remember these augmented views as well. This inspires us to use the augmented views of the original training sample to query the contrastive model to obtain multiple posteriors (one for each augmented version), and aggregate these posteriors as the input to the membership inference attack model. However, our initial attempt in this direction does not achieve a stronger attack. One reason might be our aggregation method is not optimal (we have tried averaging and concatenation). In the future, we plan to investigate more advanced aggregation operations to establish a membership inference attack tailored to contrastive models.

8 CONCLUSION

In this paper, we perform the first privacy quantification of the most representative self-supervised learning paradigm, i.e., contrastive learning. Concretely, we investigate the privacy risks of contrastive models trained on image datasets through the lens of membership inference and attribute inference. Empirical evaluation shows that contrastive models are less vulnerable to membership inference attacks compared to supervised models. This is due to the fact that contrastive models are normally less overfitted. Meanwhile, contrastive models are more prone to attribute inference attacks. We posit this is because contrastive models can generate more informative representations for data samples, which can be exploited by an adversary to achieve effective attribute inference.

To reduce the risks of attribute inference stemming from contrastive models, we propose the first privacy-preserving contrastive learning mechanism, namely *Talos*. Specifically, *Talos* introduces an adversarial classifier to censor the sensitive attributes learned by the contrastive models under the adversarial training framework. Our evaluation shows that *Talos* can effectively mitigate the attribute inference risks for contrastive models while maintaining their membership privacy and model utility.

ACKNOWLEDGMENTS

This work is partially funded by the Helmholtz Association within the project "Trustworthy Federated Data Analytics" (TFDA) (funding number ZT-I-OO1 4).

REFERENCES

[1] https://www.cs.toronto.edu/~kriz/cifar.html.

[2] https://github.com/Trusted-AI/adversarial-robustness-toolbox.

[3] Santiago Zanella Béguelin, Lukas Wutschitz, Shruti Tople, Victor Rühle, Andrew Paverd, Olga Ohrimenko, Boris Köpf, and Marc Brockschmidt. Analyzing Information Leakage of Updates to Natural Language Models. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 363–375. ACM, 2020.

[4] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion Attacks against Machine Learning at Test Time. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, pages 387–402. Springer, 2013.

[5] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting Training Data from Large Language Models. *CoRR abs/2012.07805*, 2020.

[6] Nicholas Carlini and David Wagner. Towards Evaluating the Robustness of Neural Networks. In *IEEE Symposium on Security and Privacy (S&P)*, pages 39–57. IEEE, 2017.

[7] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. GAN-Leaks: A Taxonomy of Membership Inference Attacks against Generative Models. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 343–362. ACM, 2020.

[8] Min Chen, Zhikun Zhang, Tianhao Wang, Michael Backes, Mathias Humbert, and Yang Zhang. When Machine Unlearning Jeopardizes Privacy. *CoRR abs/2005.02205*, 2020.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607. PMLR, 2020.

[10] Christopher A. Choquette Choo, Florian Tramèr, Nicholas Carlini, and Nicolas Papernot. Label-Only Membership Inference Attacks. *CoRR abs/2007.14321*, 2020.

[11] Adam Coates, Andrew Y. Ng, and Honglak Lee. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 215–223. JMLR, 2011.

[12] Maximin Coavoux, Shashi Narayan, and Shay B. Cohen. Privacy-preserving Neural Representations of Text. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1–10. ACL, 2018.

[13] Harrison Edwards and Amos J. Storkey. Censoring Representations with an Adversary. In *International Conference on Learning Representations (ICLR)*, 2016.

[14] Yanai Elazar and Yoav Goldberg. Adversarial Removal of Demographic Attributes from Text Data. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 11–21. ACL, 2018.

[15] Yaroslav Ganin and Victor S. Lempitsky. Unsupervised Domain Adaptation by Backpropagation. In *International Conference on Machine Learning (ICML)*, pages 1180–1189. JMLR, 2015.

[16] John M. Giorgi, Osvald Nitski, Bo Wang, and Gary D. Bader. DeCLUTR: Deep Contrastive Learning for Unsupervised Textual Representations. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 879–895. ACL, 2021.

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 2672–2680. NIPS, 2014.

[18] Michael Gutmann and Aapo Hyvärinen. Noise-Contrastive Estimation: A New Estimation Principle for Unnormalized Statistical Models. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 297–304. JMLR, 2010.

[19] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. LOGAN: Evaluating Privacy Leakage of Generative Models Using Generative Adversarial Networks. *Symposium on Privacy Enhancing Technologies Symposium*, 2019.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735. IEEE, 2020.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778. IEEE, 2016.

[22] Xinlei He, Jinyuan Jia, Michael Backes, Neil Zhenqiang Gong, and Yang Zhang. Stealing Links from Graph Neural Networks. In *USENIX Security Symposium (USENIX Security)*. USENIX, 2021.

[23] Xinlei He, Rui Wen, Yixin Wu, Michael Backes, Yun Shen, and Yang Zhang. Node-Level Membership Inference Attacks Against Graph Neural Networks. *CoRR abs/2102.05429*, 2021.

[24] R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. Learning Deep Representations by Mutual Information Estimation and Maximization. In *International Conference on Learning Representations (ICLR)*, 2019.

[25] Matthew Jagielski, Nicholas Carlini, David Berthelot, Alex Kurakin, and Nicolas Papernot. High Accuracy and High Fidelity Extraction of Neural Networks. In *USENIX Security Symposium (USENIX Security)*, pages 1345–1362. USENIX, 2020.

[26] Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning. In *IEEE Symposium on Security and Privacy (S&P)*, pages 19–35. IEEE, 2018.

[27] Jinyuan Jia and Neil Zhenqiang Gong. AttriGuard: A Practical Defense Against Attribute Inference Attacks via Adversarial Machine Learning. In *USENIX Security Symposium (USENIX Security)*, pages 513–529. USENIX, 2018.

[28] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. MemGuard: Defending against Black-Box Membership Inference Attacks via Adversarial Examples. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 259–274. ACM, 2019.

[29] Yizhu Jiao, Yun Xiong, Jiawei Zhang, Yao Zhang, Tianqi Zhang, and Yangyong Zhu. Sub-graph Contrast for Scalable Self-Supervised Graph Representation Learning. *CoRR abs/2009.10273*, 2020.

[30] Kalpesh Krishna, Gaurav Singh Tomar, Ankur P. Parikh, Nicolas Papernot, and Mohit Iyyer. Thieves on Sesame Street! Model Extraction of BERT-based APIs. In *International Conference on Learning Representations (ICLR)*, 2020.

[31] Klas Leino and Matt Fredrikson. Stolen Memories: Leveraging Model Memorization for Calibrated White-Box Membership Inference. In *USENIX Security Symposium (USENIX Security)*, pages 1605–1622. USENIX, 2020.

[32] Shaofeng Li, Shiqing Ma, Minhui Xue, and Benjamin Zi Hao Zhao. Deep Learning Backdoors. *CoRR abs/2007.08273*, 2020.

[33] Zheng Li and Yang Zhang. Membership Leakage in Label-Only Exposures. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 2021.

[34] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Zhaoyu Wang, Li Mian, Jing Zhang, and Jie Tang. Self-supervised Learning: Generative or Contrastive. *CoRR abs/2006.08218*, 2020.

[35] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep Learning Face Attributes in the Wild. In *IEEE International Conference on Computer Vision (ICCV)*, pages 3730–3738. IEEE, 2015.

[36] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting Unintended Feature Leakage in Collaborative Learning. In *IEEE Symposium on Security and Privacy (S&P)*, pages 497–512. IEEE, 2019.

[37] Milad Nasr, Reza Shokri, and Amir Houmansadr. Machine Learning with Membership Privacy using Adversarial Regularization. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 634–646. ACM, 2018.

[38] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning. In *IEEE Symposium on Security and Privacy (S&P)*, pages 1021–1035. IEEE, 2019.

[39] Milad Nasr, Shuang Song, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlini. Adversary Instantiation: Lower Bounds for Differentially Private Machine Learning. In *IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2021.

[40] Seong Joon Oh, Max Augustin, Bernt Schiele, and Mario Fritz. Towards Reverse-Engineering Black-Box Neural Networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[41] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff Nets: Stealing Functionality of Black-Box Models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4954–4963. IEEE, 2019.

[42] Xudong Pan, Mi Zhang, Shouling Ji, and Min Yang. Privacy Risks of General-Purpose Language Models. In *IEEE Symposium on Security and Privacy (S&P)*, pages 1471–1488. IEEE, 2020.

[43] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael Wellman. SoK: Towards the Science of Security and Privacy in Machine Learning. In *IEEE European Symposium on Security and Privacy (Euro S&P)*, pages 399–414. IEEE, 2018.

[44] Nicolas Papernot, Patrick D. McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The Limitations of Deep Learning in Adversarial Settings. In *IEEE European Symposium on Security and Privacy (Euro S&P)*, pages 372–387. IEEE, 2016.

[45] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. Scalable Private Learning with PATE. In *International Conference on Learning Representations (ICLR)*, 2018.

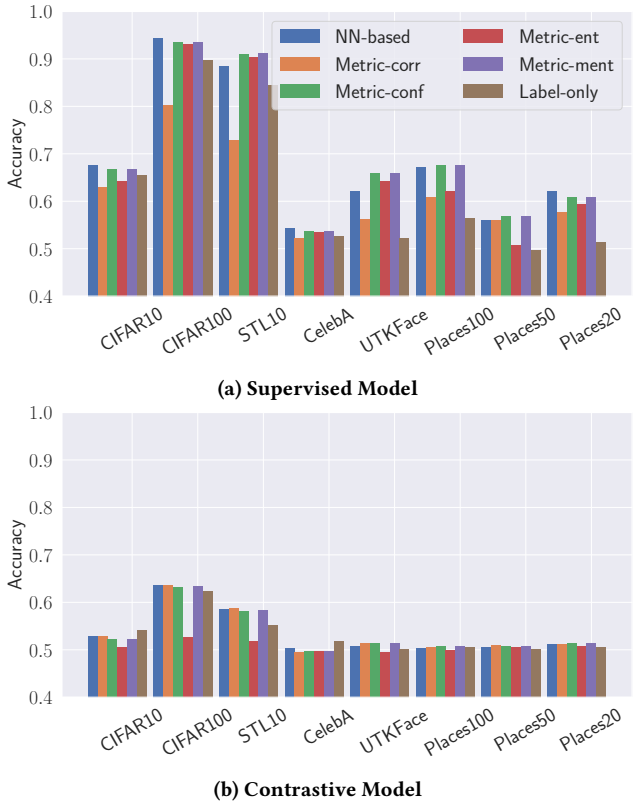
[46] Nisarg Raval, Ashwin Machanavajjhala, and Jerry Pan. Olympus: Sensor Privacy through Utility Aware Obfuscation. *Symposium on Privacy Enhancing Technologies Symposium*, 2019.

[47] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *CoRR abs/1409.0575*, 2015.

[48] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. Updates-Leak: Data Set Inference and Reconstruction Attacks in Online

Learning. In *USENIX Security Symposium (USENIX Security)*, pages 1291–1308. USENIX, 2020.
[49] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2019.
[50] Mark Sandler, Andrew G. Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520. IEEE, 2018.
[51] Roei Schuster, Congzheng Song, Eran Tromer, and Vitaly Shmatikov. You Autocomplete Me: Poisoning Vulnerabilities in Neural Code Completion. *CoRR abs/2007.02220*, 2020.
[52] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership Inference Attacks Against Machine Learning Models. In *IEEE Symposium on Security and Privacy (S&P)*, pages 3–18. IEEE, 2017.
[53] Congzheng Song and Ananth Raghunathan. Information Leakage in Embedding Models. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 377–390. ACM, 2020.
[54] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. Machine Learning Models that Remember Too Much. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 587–601. ACM, 2017.
[55] Congzheng Song and Vitaly Shmatikov. Auditing Data Provenance in Text-Generation Models. In *ACM Conference on Knowledge Discovery and Data Mining (KDD)*, pages 196–206. ACM, 2019.
[56] Congzheng Song and Vitaly Shmatikov. Overlearning Reveals Sensitive Attributes. In *International Conference on Learning Representations (ICLR)*, 2020.
[57] Liwei Song and Prateek Mittal. Systematic Evaluation of Privacy Risks of Machine Learning Models. In *USENIX Security Symposium (USENIX Security)*. USENIX, 2021.
[58] Liwei Song, Reza Shokri, and Prateek Mittal. Privacy Risks of Securing Machine Learning Models against Adversarial Examples. In *ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 241–257. ACM, 2019.
[59] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. Ensemble Adversarial Training: Attacks and Defenses. In *International Conference on Learning Representations (ICLR)*, 2017.
[60] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing Machine Learning Models via Prediction APIs. In *USENIX Security Symposium (USENIX Security)*, pages 601–618. USENIX, 2016.
[61] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. *CoRR abs/1807.03748*, 2018.
[62] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 2008.
[63] Binghui Wang and Neil Zhenqiang Gong. Stealing Hyperparameters in Machine Learning. In *IEEE Symposium on Security and Privacy (S&P)*, pages 36–52. IEEE, 2018.
[64] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised Feature Learning via Non-Parametric Instance Discrimination. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3733–3742. IEEE, 2018.
[65] Qizhe Xie, Zihang Dai, Yulun Du, Eduard H. Hovy, and Graham Neubig. Controllable Invariance through Adversarial Feature Learning. In *Annual Conference on Neural Information Processing Systems (NIPS)*, pages 585–596. NIPS, 2017.
[66] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting. In *IEEE Computer Security Foundations Symposium (CSF)*, pages 268–282. IEEE, 2018.
[67] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph Contrastive Learning with Augmentations. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*. NeurIPS, 2020.
[68] Zhifei Zhang, Yang Song, and Hairong Qi. Age Progression/Regression by Conditional Adversarial Autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4352–4360. IEEE, 2017.
[69] Bolei Zhou, Àgata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

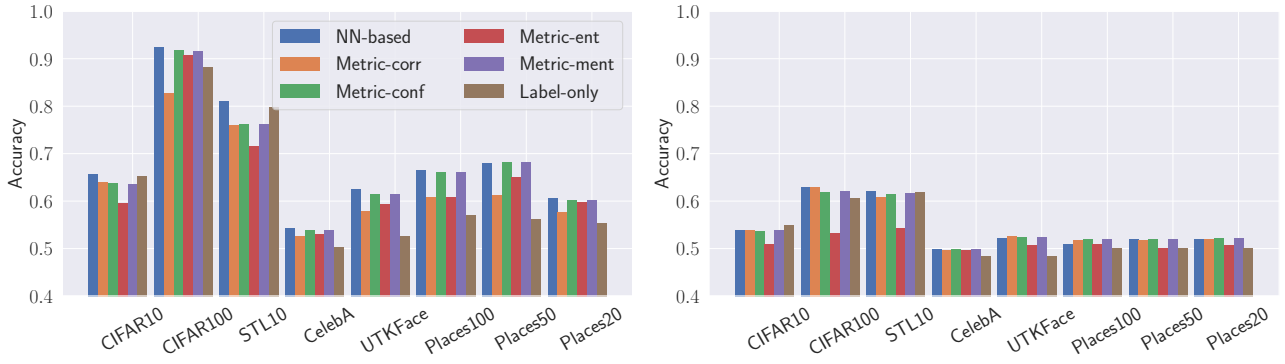
A APPENDIX



(a) Supervised Model

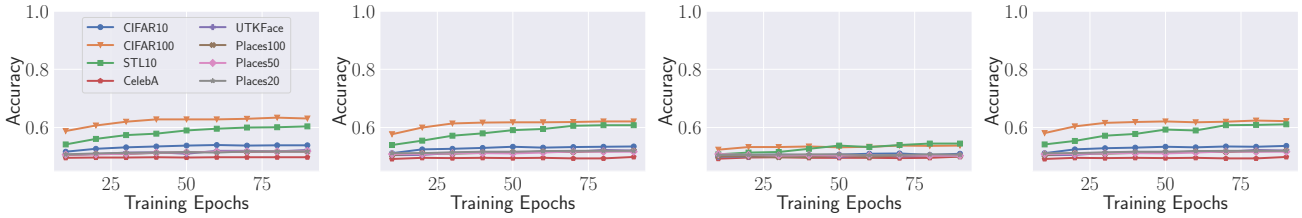
(b) Contrastive Model

Figure 15: The performance of different membership inference attacks against both supervised models and contrastive models with ResNet-18 on 8 different datasets. The x-axis represents different datasets. The y-axis represents membership inference attacks' accuracy.



(a) Supervised Model (b) Contrastive Model

Figure 16: The performance of different membership inference attacks against both supervised models and contrastive models with ResNet-50 on 8 different datasets. The x-axis represents different datasets. The y-axis represents membership inference attacks' accuracy.

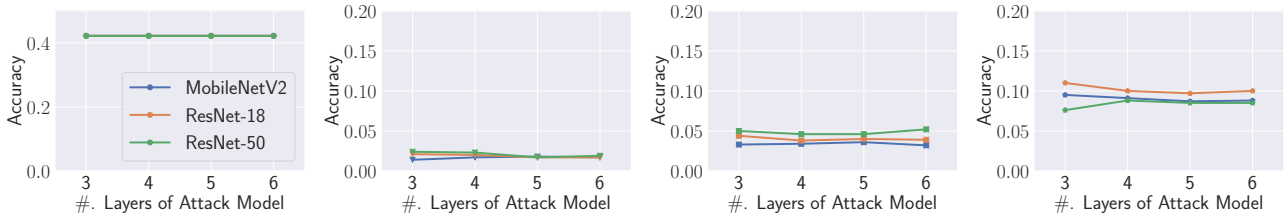


(a) Metric-corr (b) Metric-conf (c) Metric-ent (d) Metric-ment

Figure 17: The performance of metric-based membership inference attacks against contrastive models with ResNet-50 on 8 different datasets under different numbers of epochs for classification layer training. The x-axis represents different numbers of epochs. The y-axis represents membership inference attacks' accuracy. Each line corresponds to a specific dataset.

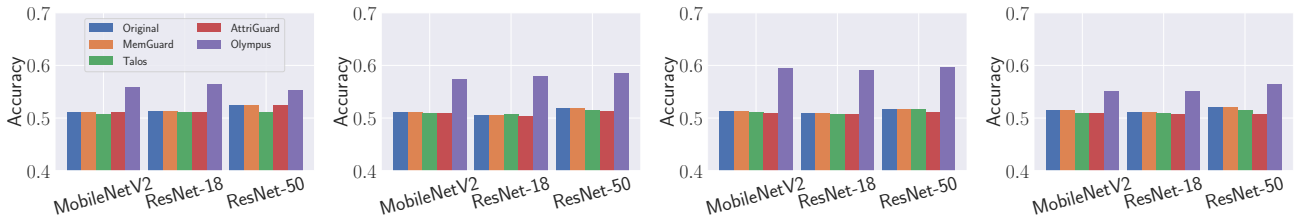
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 18: The performance of attribute inference attacks against supervised models on 4 different datasets under different percentages of the attack training dataset. The x-axis represents different percentages of the attack training dataset. The y-axis represents attribute inference attacks' accuracy.



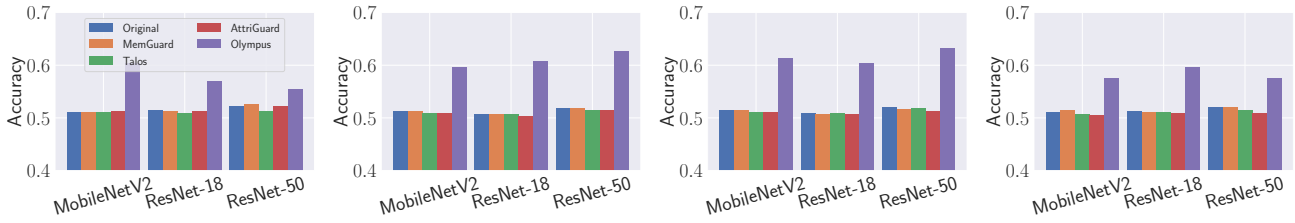
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 19: The performance of attribute inference attacks against supervised models on 4 different datasets under attack models with different layers. The x-axis represents attack models' layers. The y-axis represents attribute inference attacks' accuracy.



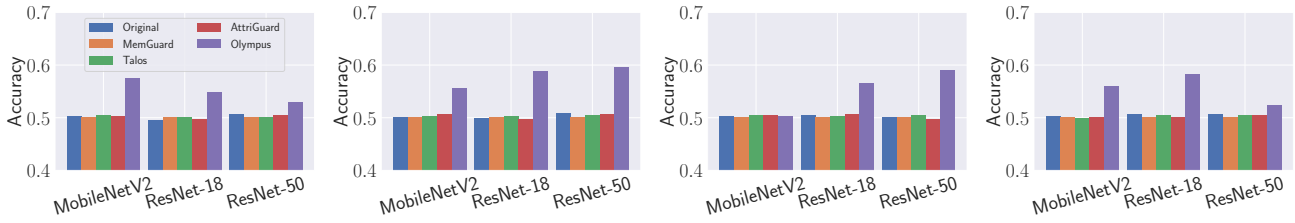
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 20: The performance of metric-corr membership inference attacks against original contrastive models,*Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of metric-corr membership inference attacks.



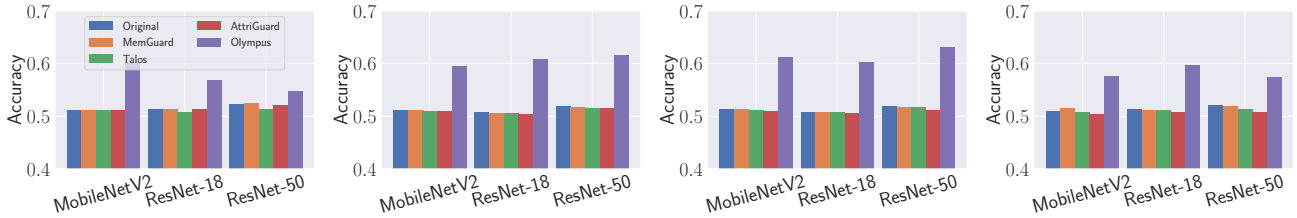
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 21: The performance of metric-conf membership inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different methods. The y-axis represents the accuracy of metric-conf membership inference attacks.



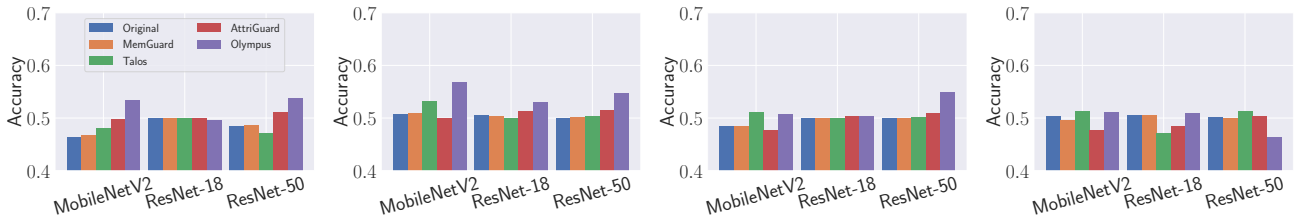
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 22: The performance of metric-ent membership inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of metric-ent membership inference attacks.



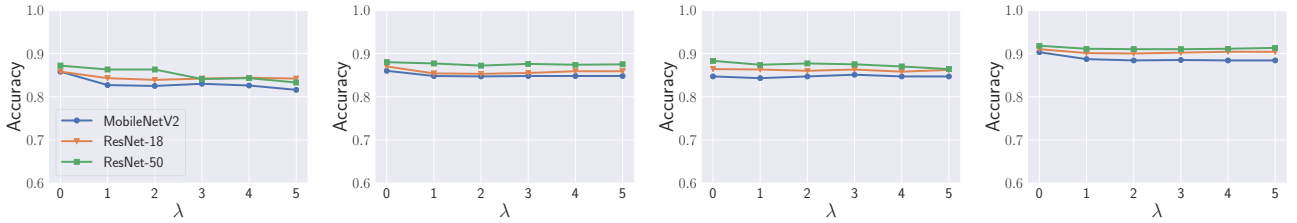
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 23: The performance of metric-ment membership inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of metric-ment membership inference attacks.



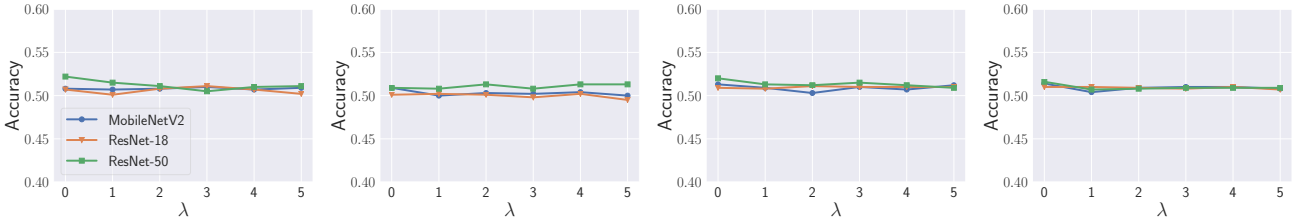
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 24: The performance of label-only membership inference attacks against original contrastive models, *Talos*, *MemGuard*, *Olympus*, and *AttriGuard* with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets. The x-axis represents different models. The y-axis represents the accuracy of label-only membership inference attacks.



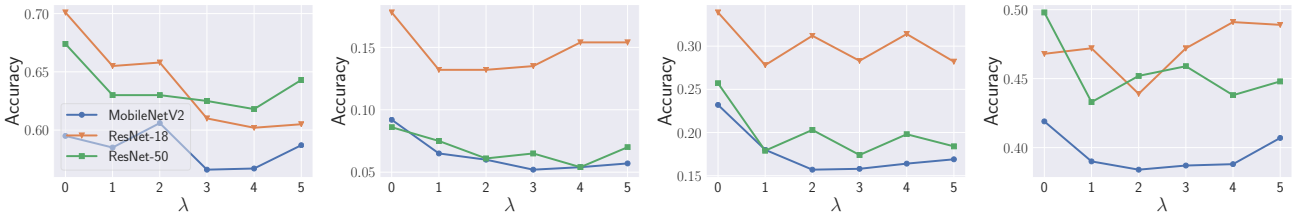
(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 25: The performance of original classification tasks for the *Talos* models with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets under different adversarial factor λ. The x-axis represents different λ. The y-axis represents the corresponding performance.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 26: The performance of membership inference attacks for the *Talos* models with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets under different adversarial factor λ. The x-axis represents different λ. The y-axis represents the corresponding performance.



(a) UTKFace (b) Places100 (c) Places50 (d) Places20

Figure 27: The performance of attribute inference attacks for the *Talos* models with MobileNetV2, ResNet-18, and ResNet-50 on 4 different datasets under different adversarial factor λ. The x-axis represents different λ. The y-axis represents the corresponding performance.