entropy

MDPI

Article

Privacy-Preserving Image Template Sharing Using Contrastive Learning

Shideh Rezaeifar *, **Slava Voloshynovskiy**, **Meisam Asgari Jirhandeh** and **Vitality Kinakh**

Department of Computer Science, University of Geneva, 1227 Carouge, Switzerland; svolos@unige.ch (S.V.); meisam.asgarijirhandeh@students.unibe.ch (M.A.J.); vitality.kinakh@unige.ch (V.K.)
* Correspondence: shideh.rezaeifar@unige.ch

Abstract: With the recent developments of Machine Learning as a Service (MLaaS), various privacy concerns have been raised. Having access to the user's data, an adversary can design attacks with different objectives, namely, reconstruction or attribute inference attacks. In this paper, we propose two different training frameworks for an image classification task while preserving user data privacy against the two aforementioned attacks. In both frameworks, an encoder is trained with contrastive loss, providing a superior utility-privacy trade-off. In the reconstruction attack scenario, a supervised contrastive loss was employed to provide maximal discrimination for the targeted classification task. The encoded features are further perturbed using the obfuscator module to remove all redundant information. Moreover, the obfuscator module is jointly trained with a classifier to minimize the correlation between private feature representation and original data while retaining the model utility for the classification. For the attribute inference attack, we aim to provide a representation of data that is independent of the sensitive attribute. Therefore, the encoder is trained with supervised and private contrastive loss. Furthermore, an obfuscator module is trained in an adversarial manner to preserve the privacy of sensitive attributes while maintaining the classification performance on the target attribute. The reported results on the CelebA dataset validate the effectiveness of the proposed frameworks.

Keywords: privacy; reconstruction attack; re-identification attack

check for updates

Citation: Rezaeifar, S.; Voloshynovskiy, S.; Asgari Jirhandeh, M.; Kinakh, V. Privacy-Preserving Image Template Sharing Using Contrastive Learning. *Entropy* **2022**, *24*, 643. https://doi.org/10.3390/e24050643

Academic Editors: Eirik Rosnes, Alexandre Graell i Amat and Hsuan-Yin Lin

Received: 4 January 2022
Accepted: 20 April 2022
Published: 3 May 2022

1. Introduction

Deep learning has been widely applied in many computer vision applications in recent years, with remarkable success. Much progress in deep learning has been made possible thanks to accessible computational power and the widely available datasets needed for training. The necessity of memory and computational power has incentivized many companies such as AMAZON, Google, and IBM to provide their customers with platforms offering Machine Learning as a Service (MLaaS). MLaaS runs on a cloud environment and covers most infrastructure issues such as data pre-processing, model training, and model evaluation. Hence, the users can deploy their machine learning models by simply uploading their data (e.g., images) into the cloud server.

With all the promises made by MLaaS, this scheme introduces various privacy challenges for both users and the service provider. From one point of view, the service providers are concerned that an adversary could be disguised as a client to steal their model parameters. On the other hand, users are worried that sensitive information might be revealed to unauthorized third parties by uploading their raw data into the cloud server [1]. Furthermore, in some financial or medical data applications, it might not be legally allowed for the user to upload and submit raw data to the cloud server. One widely used solution is to share a feature representation of data instead. However, the adversary can still exploit the privacy leakage in the feature representation and design attacks targeting various objectives.

There are mainly two types of attacks regarding the privacy of users' data: attribute inference attack and reconstruction attack [1,2]. In the reconstruction or model inversion attack, the adversary's goal is to reconstruct the original data given the shared feature representation. Whereas in attribute inference attack, the adversary is interested in identifying certain sensitive attributes in the data such as age, gender, race, etc.

In this paper, we consider an image classification task in which users send their original data to the cloud service provider. The adversary, a malicious user or the MLaaS provider, wishes to exploit the privacy leakage in the shared feature representation targeting reconstruction or attribute inference attack.

The rest of the paper is organized as follows: Problem formulation and assumptions are introduced in Section 2. Section 3 reviews the related work. Two defense frameworks against the reconstruction attack and attribute inference attack are proposed in Sections 4 and 5, respectively. Finally, Section 6 concludes this work along with suggestions for future work.

2. Problem Formulation

As shown in Figure 1, given the high dimensional images in the dataset $\mathbf{x} \in \mathcal{R}^n$, users or data owners intend to share a feature representation $\mathbf{h}$ for the specific utility task, image classification. Let $\mathbf{Y}_t$ denote the corresponding labels for the target class that the central classifier is trained to predict them and let $\mathbf{Y}_p$ denote the label information for the private and sensitive attribute. Concerned about the privacy leakage in the shared representations, the users, as the defenders, apply an obfuscation mechanism on the shared features before releasing them to the public as $\mathbf{h}_p$. The defender's ultimate goal is to maintain a good classification performance while preserving their privacy.

On the other hand, having access to a collection of original images and their corresponding protected features $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{h}_{p_1}), (\mathbf{x}_2, \mathbf{h}_{p_2}), \dots \mathbf{x}_N, \mathbf{h}_{p_N})\}$, the adversary aims to reconstruct the original data or recognize sensitive attributes such as age, gender, etc. Therefore, in this setting, the utility is a classification task and privacy is defined as the attacker's ability to reconstruct the original data or re-identify the sensitive attributes.

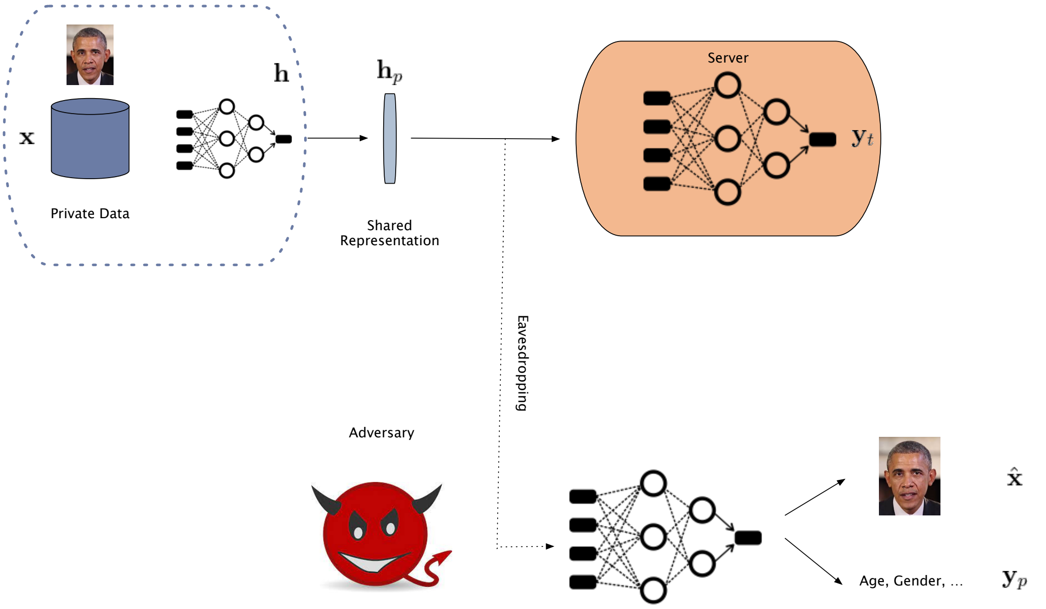


Figure 1. Threat model. The user sends the private representations to the server for final classification. Eavesdropping on the private features, the adversary wishes to reconstruct the original data or infer sensitive attributes. The adversary does not have access to the local obfuscation mechanism used by the user, shown in blue dashed lines.

3. Related Work

Several techniques have been introduced to preserve the users' data privacy, such as image obfuscation, homomorphic encryption, secure multi-party computation, and private feature representation.

Classical image obfuscation: In image obfuscation techniques, the original image is perturbed to hide sensitive information or details and make it visually unidentifiable. Conventional methods include pixelating [3], blurring [3,4], and masking [5]. However, as discussed in [6,7], these protected images can still be identified or reconstructed using deep learning-assisted methods. Recently, more advanced frameworks of deep obfuscation based on deep generative models have been introduced [8–10].

Homomorphic encryption: Homomorphic encryption (HE) is another method that allows one to carry out computations on encrypted data without the need for decryption [11]. This means that data can be processed securely even though they have been outsourced in untrusted and public environments. HE can be categorized into three types, namely partially homomorphic (PHE), somewhat homomorphic (SWHE), and fully homomorphic encryption (FHE) [11]. However, the operations in HE are limited to be represented as a polynomial of a bounded degree. They cannot, therefore, be used with complicated and nonlinear computation functions. Moreover, HE is highly computationally intensive and leads to an extremely slow training process.

Deep and private feature sharing: With the recent advancements of deep models, a new line of work has been introduced to share deep private and obfuscated feature representations of images. Osia et al. [12] considered a client-server setting in which the deep model architecture is separated into two parts: a feature extractor on the client's side and a classifier on the cloud. The extracted features are then protected against attribute inference attacks by adding noise and Siamese fine-tuning. However, their proposed framework is not feasible during training due to its interactive training procedure and high communication throughput between the clients and servers [13].

Later, Li et al. proposed PrivyNet, a private deep learning training framework [13]. PrivyNet splits a neural network into local and cloud counterparts. The feature representations of private data are extracted using the local model while the cloud neural network is trained on publicly released features for the target classification task. The authors considered a reconstruction attack on the shared features and measured privacy through the reconstruction error. In ref. [14], the authors used an adversarial training scheme between an encoder and a classifier to preserve the privacy of intermediate encoded features from attribute inference attacks.

Along the same line of research, Lie et al. [15] introduced an adversarial privacy network called PAN to learn obfuscated features. The learned that obfuscated features are designed to be effective against both reconstruction attacks and attribute inference attacks. Similarly, DeepObfuscater was introduced in ref. [16], and the authors extended PAN to include perceptual quality.

In the context of privacy of published datasets, Huang et al. [17] proposed a framework based on a minmax game between a privatizer and an adversary. By employing generative adversarial networks (GAN) in their framework, users can directly learn to privatize their dataset without having access to the dataset statistics.

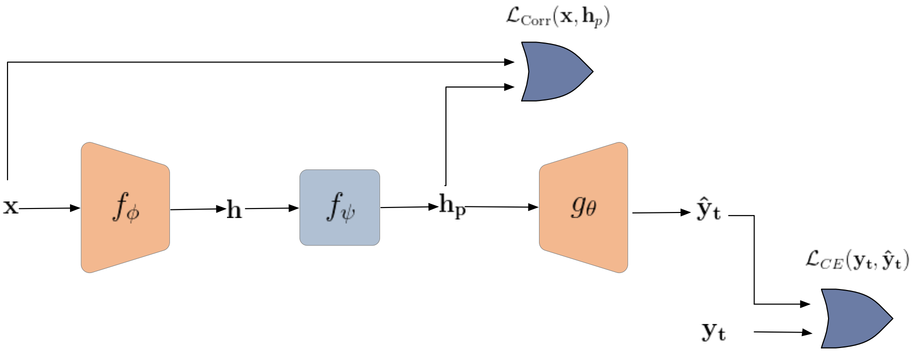
4. Defense Against a Reconstruction Attack

This section introduces a framework to maintain a good classification accuracy while avoiding the invertibility of shared representations. In other words, the proposed framework is designed to keep only relevant information for the specific classification task. The model consists of three modules: encoder, obfuscator, and classifier. The encoder is trained using supervised contrastive loss to provide maximal discrimination for the classification task. The encoded features are obfuscated by minimizing their statistical correlation to the original input images. Finally, a classifier is jointly trained to maintain the classification performance.

4.1. *Proposed Architecture*

The overall private data-sharing framework, shown in Figure 2, consists of three steps:

1. An *encoder* f_ϕ is pre-trained on the public data using supervised contrastive loss. The encoder is later used to extract discriminative representation for the targeted classification task;
2. An *obfuscator* f_ψ is learned to remove irrelevant information in representation $\mathbf{h}$ by minimizing its correlation to the original data $\mathbf{x}$;
3. A *classifier* g_θ is jointly trained with the obfuscator to ensure that the useful information for the intended classification task is preserved in the obfuscated representation.



$\mathcal{L}_{\text{Corr}}(\mathbf{x}, \mathbf{h}_p)$
x
f_ϕ
h
f_ψ
$\mathbf{h_p}$
g_θ
$\hat{\mathbf{y}}_\mathbf{t}$
$\mathcal{L}_{CE}(\mathbf{y_t}, \hat{\mathbf{y}}_\mathbf{t})$
$\mathbf{y_t}$

Figure 2. General diagram of the proposed framework for defense against reconstruction attack. $\mathcal{L}_{CE}$ denotes cross-entropy and $\mathcal{L}_{\text{Corr}}$ stands for a similarity metric.

4.1.1. Encoder

As shown in Figure 3, the encoder f_ϕ is initially trained with a contrastive loss to output a well-discriminated feature representation. To this end, we used a ResNet backbone with contrastive loss similar to the SimCLR approach [18].

The basic idea behind contrastive learning is to pull similar instances denoted as positive pairs together and push dissimilar ones, negative samples, apart. Given a random augmentation transform $\mathcal{T}_t(.)$, two different views $\mathbf{x}_i, \mathbf{x}_j$ of the same image $\mathbf{x}$ are considered as positive pairs, and the rest of the batch samples as negative pairs. A projection head $g_\theta(.)$ maps the feature representations of the base encoder to the latent embedding $\mathbf{z}$ [18]:

$$\begin{aligned} \mathbf{x}_i &= \mathcal{T}_{t_i}(\mathbf{x}), \quad \mathbf{h}_i = f_\phi(\mathbf{x}_i), \quad \mathbf{z}_i = g_\theta(\mathbf{h}_i); \\ \mathbf{x}_j &= \mathcal{T}_{t_j}(\mathbf{x}), \quad \mathbf{h}_j = f_\phi(\mathbf{x}_j), \quad \mathbf{z}_j = g_\theta(\mathbf{h}_j). \end{aligned} \tag{1}$$

Using cosine similarity, the similarities between positive pairs are maximized while the negative ones are minimized. The self-supervised contrastive loss is defined as:

$$\mathcal{L}_{ssl} = -\sum_i \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_j))}{\sum_{k,k\neq i} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k))}. \tag{2}$$

This idea was further extended to include target class information in the loss where feature representations from the same class are pulled closer together than those from different classes [19].

$$\mathcal{L}_{supcon} = -\sum_i \frac{1}{|P(i)|} \sum_{p\in P(i)} \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_p))}{\sum_{k,k\neq i} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k))}, \tag{3}$$

where $P(i)$ are all the positive samples belonging to the same class as $\mathbf{x}_i$.

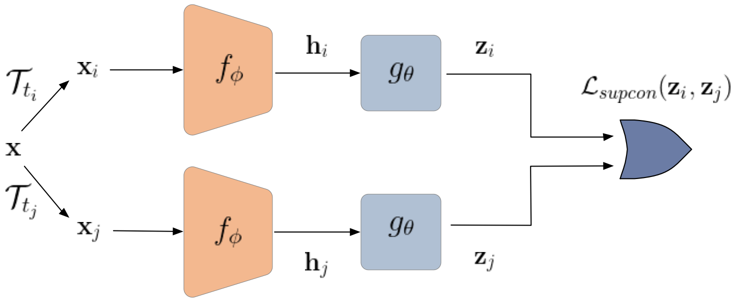


Figure 3. Encoder training using supervised contrastive learning.

4.1.2. Obfuscator

The obfuscator f_ψ is trained to avoid the invertibility of shared feature representation. From an information-theoretic point of view, $\mathbf{X} \rightarrow \mathbf{H} \rightarrow \hat{\mathbf{X}}$ forms a Markov chain. To mitigate the reconstruction attack, $I(\mathbf{X}, \hat{\mathbf{X}})$ should be minimized. A widely used approach is to jointly train an adversary image decoder to achieve reconstruction disparity by minimizing the Structural Similarity Index Measure (SSIM) [20]. This is done through a min-max optimization game between the obfuscator and adversary decoder.

Nevertheless, considering the information processing inequality based on the above Markov chain, minimizing the mutual information between the original image $\mathbf{X}$ and the feature representation $\mathbf{H}$ upper bounds the $I(\mathbf{X}, \hat{\mathbf{X}})$ as $I(\mathbf{X}, \mathbf{H}) \geq I(\mathbf{X}, \hat{\mathbf{X}})$.

To minimize $I(\mathbf{X}, \mathbf{H})$, one should estimate the mutual information, which is a well-known and challenging problem and would involve a more complicated optimization. To solve this issue and to accelerate and simplify the training, we adopted two statistical correlation measures between random variables, namely, Hilbert–Schmidt Independence Criterion (HSIC) [21,22] and Distance Correlation (DistCorr) [23]. Consequently, the obfuscator network f_ψ is trained to minimize the correlation between the original images and the protected representation:

$$\mathcal{L}_{\text{Corr}} = \text{Corr}(\mathbf{x}, \mathbf{h_p}), \tag{4}$$

where Corr(.) can be either based on distance correlation DistCorr or Hilbert–Schmidt Independence Criterion HSIC. The idea of minimizing the statistical dependencies of features has been around in the literature of federated or distributed learning and physics [24–26].

Hilbert–Schmidt Independence Criterion (HSIC): Let $\mathcal{F}$ be a reproducing kernel Hilbert space (RKHS), with the continuous feature mapping $\phi(\mathbf{x})$ and kernel function $k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$. Similarly, assume $\mathcal{G}$ be an RKHS, with the continuous feature mapping $\psi(\mathbf{h})$ and kernel function $k(\mathbf{h}, \mathbf{h}') = \langle \psi(\mathbf{h}), \psi(\mathbf{h}') \rangle$.

The cross-covariance operator $\mathbf{C_{xh}} : \mathcal{G} \rightarrow \mathcal{F}$ can be defined as [21,22]:

$$\mathbf{C_{xh}} := \mathbf{E}_{\mathbf{p}(\mathbf{x},\mathbf{h})}[(\phi(\mathbf{x}) - \mu_\mathbf{x}) \otimes (\psi(\mathbf{h}) - \mu_\mathbf{h}], \tag{5}$$

where $\otimes$ is the matrix product and $\mu_\mathbf{x} = \mathbf{E}_{\mathbf{p}(\mathbf{x})}[\phi(\mathbf{x})]$, $\mu_\mathbf{h} = \mathbf{E}_{\mathbf{p}(\mathbf{h})}[\psi(\mathbf{h})]$. The largest singular value of the cross-covariance operator $\|\mathbf{C_{xh}}\|$ is zero if and only if $\mathbf{x}$ and $\mathbf{h}$ are independent

The Hilbert–Schmidt Independence Criterion is defined as the squared Hilbert–Schmidt norm of the associated cross-covariance operator $\mathbf{C_{xh}}$:

$$\text{HSIC}_{x,h}(\mathcal{F}, \mathcal{G}) = \|\mathbf{C_{xh}}\|^2_{HS}. \tag{6}$$

Distance Correlation (DistCorr): Let $\mathbf{X}$ and $\mathbf{H}$ be two random vectors with finite second moments. Assume that $(\mathbf{X}, \mathbf{H})$, $(\mathbf{X}', \mathbf{H}')$, $(\mathbf{X}'', \mathbf{H}'')$ are independent and identically distributed. Then, the distance covariance can be defined as:

$$\begin{aligned} \text{dCov}(\mathbf{X}, \mathbf{H}) =& \mathbf{E}(|\mathbf{X} - \mathbf{X}'||\mathbf{H} - \mathbf{H}'|) \\ &+ \mathbf{E}(|\mathbf{X} - \mathbf{X}'|)\mathbf{E}(|\mathbf{H} - \mathbf{H}'|) \\ &- 2\mathbf{E}(|\mathbf{X} - \mathbf{X}'||\mathbf{H} - \mathbf{H}''|), \end{aligned} \tag{7}$$

where $|.|$ is the pairwise distance. Subsequently, the definition of the distance correlation will be:

$$\text{DistCorr}(\mathbf{X}, \mathbf{H}) = \frac{\text{dCov}(\mathbf{X}, \mathbf{H})}{\sqrt{\text{dCov}(\mathbf{X}, \mathbf{X})\, \text{dCov}(\mathbf{H}, \mathbf{H})}}. \tag{8}$$

4.1.3. Classifier

The classifier g_θ is a lightweight neural network with two fully connected layers and Relu activation functions. The classifier is jointly trained with the obfuscator to maintain the classification accuracy for the utility task:

$$(\hat{\theta}, \hat{\psi}) = \text{argmin}_{\theta,\psi}\, \mathcal{L}_{CE}(\mathbf{y}_t, \hat{\mathbf{y}}_t) + \gamma \mathcal{L}_{\text{Corr}}(\mathbf{x}, \mathbf{h}_p), \tag{9}$$

where γ is the utility-privacy trade-off parameter. $\mathcal{L}_{CE}$ denotes the cross-entropy between the utility attribute $\mathbf{y}_t$ and its estimate $\hat{\mathbf{y}}_t$ and $\mathcal{L}_{\text{Corr}}$ denotes either DistCorr or HSIC according to Equations (6) and (8).

4.2. Experimental Results

4.2.1. Experimental Setup

Dataset: We conducted experiments on a celebrity face image dataset, CelebA [27], which consists of over 20,000 celebrity images, where each image is annotated with 40 attributes. Every input image is center-cropped by 178 × 178 and then resized to 128 × 128. We select the "gender" attribute for our intended classification task.

Attacker setup: The adversary has a set of publicly available protected representations $\mathbf{h}_p$ with the corresponding original images $\mathbf{x}$ and aims to train a decoder to reconstruct the original image for the model inversion attack.

4.2.2. Visualizations of Encoded Features

This section investigates the effect of using supervised contrastive loss in the encoded features. To do so, we visualized the 2D t-SNEs of extracted features for the target class label of "gender," as depicted in Figure 4. As expected, the output features of the encoder trained with supervised loss are more discriminative compared to those trained in the unsupervised way.

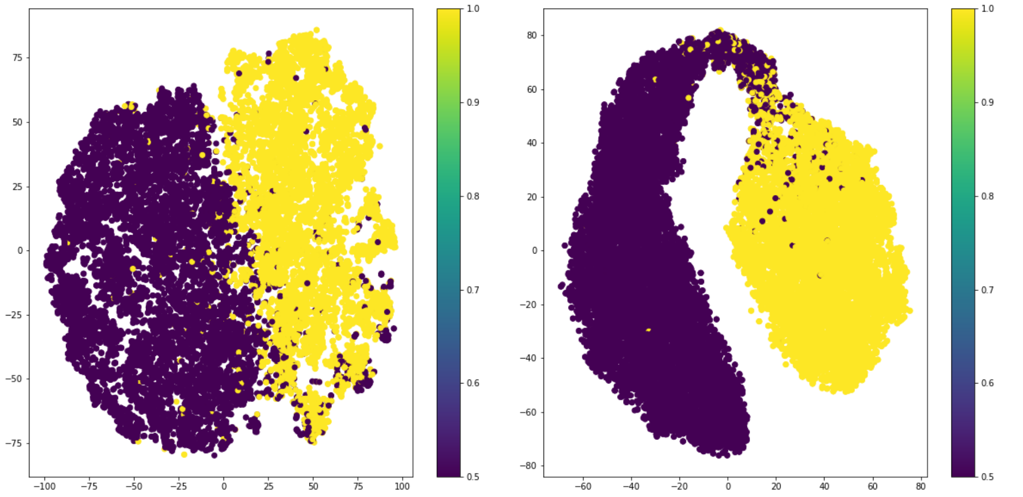


Figure 4. T-SNE visualization of output features for unsupervised and supervised contrastive losses.

4.2.3. Classification Performance

In this section, the utility-privacy trade-off is investigated in the form of classification accuracy vs. decorrelation. More specifically, we are interested in analyzing the extent to which classification accuracy decreases if we decorrelate the features from original images. As shown in Table 1, with only 0.2 loss in the accuracy, the correlation between input images and the features drops for both similarity measures. In the case of HSIC, however, the reduction in correlation is remarkable. The considerably smaller loss in the accuracy is mainly due to the supervised contrastive loss used in training the encoder, as we obtain discriminative features with respect to the target class. In Section 4.2.4, we demonstrate that an attacker can still reconstruct completely recognizable images using these discriminative features. Consequently, the obfuscator aims at removing all the redundant information about the images and only keeping the ones related to the intended classification task.

Table 1. Classification vs. Correlation.

Correlation Type	Accuracy	DistCorr	HSIC
without	98.48	0.714	0.62
DistCorr, $\gamma = 2$	98.2	0.24	0.25
DistCorr, $\gamma = 20$	98.1	0.21	0.23
HSIC, $\gamma = 2$	98.23	0.32	0.026
HSIC, $\gamma = 20$	98.17	0.29	0.007

4.2.4. Reconstruction Attack

According to Figure 5, the adversary model for the reconstruction attack consists of a generator G_{θ_x} and a discriminator $D_{\theta_{x\hat{x}}}$. The generator network maps the protected and obfuscated feature representation $\mathbf{h}_p$ to the image space, while the discriminator evaluates them. The discriminator network assigns a probability that the image is from the real data distribution rather than the generator distribution. Thus, the discriminator is trained to classify images as being from the training data or reconstructed from the generator:

$$\mathcal{L}_D = \log(D_{\theta_{x\hat{x}}}(\mathbf{x})) + \log(1 - D_{\theta_{x\hat{x}}}(G_{\theta_x}(\mathbf{h}_p))). \tag{10}$$

Therefore, the decoder and generator are trained in a min-max optimization problem:

$$\min_{g_x} \max_{\theta_{x\hat{x}}} \mathbf{E}_{p(\mathbf{x})}[\log(D_{\theta_{x\hat{x}}}(\mathbf{x}))] + \mathbf{E}_{p(\mathbf{h}_p)}[\log(1 - D_{\theta_{x\hat{x}}}(G_{\theta_x}(\mathbf{h}_p))]. \tag{11}$$

To improve the performance of the generator, a perceptual loss similar to SRGAN [28] was also employed. The perceptual loss for the generator network consists of an adversarial loss and a content loss:

$$\mathcal{L}_{\text{perceptual}} = \underbrace{\mathcal{L}_{\text{mse}} + \mathcal{L}_{\text{vgg}}}_{\text{content loss}} + \underbrace{\mathcal{L}_{D_g}}_{\text{adversarial loss}}, \tag{12}$$

and:

$$\begin{aligned}
\mathcal{L}_{\text{mse}} &= \mathbf{E}_{p(\mathbf{x},\mathbf{h}_p)} \|\mathbf{x} - G_{\theta_x}(\mathbf{h}_p)\|, \\
\mathcal{L}_{\text{vgg}} &= \mathbf{E}_{p(\mathbf{x},\mathbf{h}_p)} \|\,\text{vgg}_{19}(\mathbf{x}) - \text{vgg}_{19}(G_{\theta_x}(\mathbf{h}_p))\|, \\
\mathcal{L}_{D_g} &= \mathbf{E}_{p(\mathbf{h}_p)}[-\log(D_{\theta_{x\hat{x}}}(G_{\theta_x}(\mathbf{h}_p)))],
\end{aligned} \tag{13}$$

where $\text{vgg}_{19}(.)$ is the output of a pre-trained 19-layer VGG network [29].

We conducted experiments on the reconstruction attack for different correlation losses and different values of γ in Equation (9). The performance of the attack model is evaluated using multi-scale structural similarity (MSSIM) [30] and SSIM [20]. To better evaluate the effectiveness of the proposed obfuscation model, the reconstruction quality from the following scenarios has been considered:

- $\mathbf{h}$: The feature representations of original images;
- $\mathbf{h}_{x_{\text{noisy}}}$: The raw images are perturbed by adding Gaussian noise and fed to the encoder to get the features;
- $\mathbf{h}_{\text{noisy}}$: The feature representations of original images are perturbed by adding Gaussian noise;
- $\mathbf{h}_p$: The obfuscated and protected features.

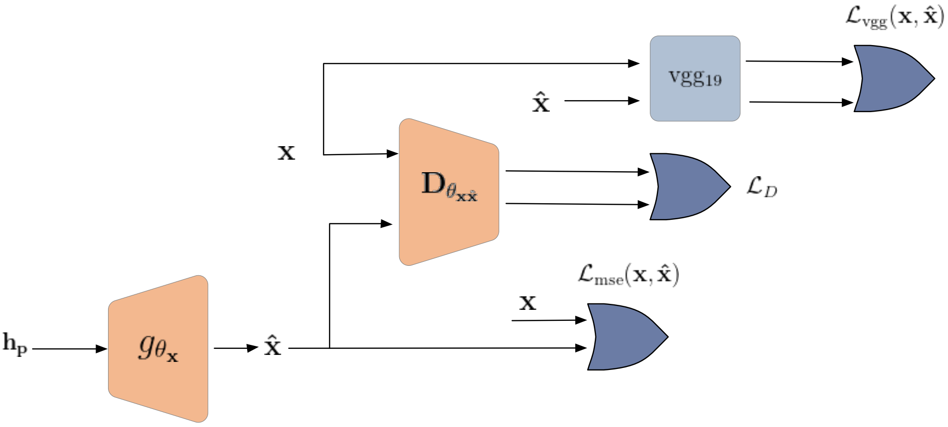

$\mathcal{L}_{\text{vgg}}(\mathbf{x}, \hat{\mathbf{x}})$
vgg19
$\hat{\mathbf{x}}$
$\mathbf{x}$
$\mathbf{D}_{\theta_{\mathbf{x}\hat{\mathbf{x}}}}$
$\mathcal{L}_D$
$\mathcal{L}_{\text{mse}}(\mathbf{x}, \hat{\mathbf{x}})$
$\mathbf{x}$
$\mathbf{h}_\text{p}$
$g_{\theta_\mathbf{x}}$
$\hat{\mathbf{x}}$

Figure 5. Adversary model for reconstruction attack.

The average SSIM and MSSIM for reconstructed images from the protected features and three other scenarios are reported in Table 2. As the SSIM and MSSIM scores were very close for both correlation measures and different values of γ, we only reported the one for DistCorr and $\gamma = 2$ in Table 2. The results show that both similarity measures are dropped by a large margin with only a 0.2% loss in accuracy, therefore validating the effectiveness of the obfuscator.

Table 2. Image reconstruction comparison.

Obfuscation	SSIM	MSSIM	Accuracy
$\mathbf{h}$	0.4	0.56	98.48
$\mathbf{h}_{x_{\text{noisy}}}$	0.36	0.50	98.41
$\mathbf{h}_{\text{noisy}}$	0.30	0.43	98.37
$\mathbf{h}_p$	0.19	0.16	98.2

Moreover, the visualization of the reconstructed images is illustrated in Figure 6. The reconstructed images from the raw features are completely recognizable, but not very sharp. This is mainly because the encoder is trained with the supervised contrastive loss, where the information about the target class is mostly left in the last layer. On the other hand, the output images become completely unrecognizable with our proposed obfuscator, and even a powerful decoder can only output an average image. To further investigate the effect of correlation measure and γ in Equation (9), the output images for different cases are presented in Figure 7. Even though the attacker outputs an average image for both cases of correlation measures, it is interesting to note that features learned by HSIC produce different average images for males and females. In other words, the gender information is clearly preserved in the protected representation.

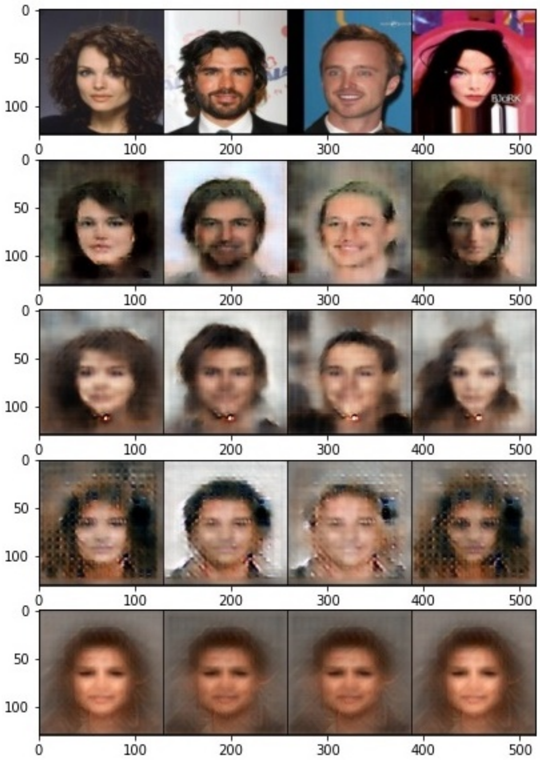


Figure 6. Visual performance of the reconstruction attack from different features. First row: $\mathbf{h}$, second row: $\mathbf{h}_{x_{\text{noisy}}}$, third row: $\mathbf{h}_{\text{noisy}}$, and the last row: $\mathbf{h}_p$ for DistCorr, $\gamma = 20$.

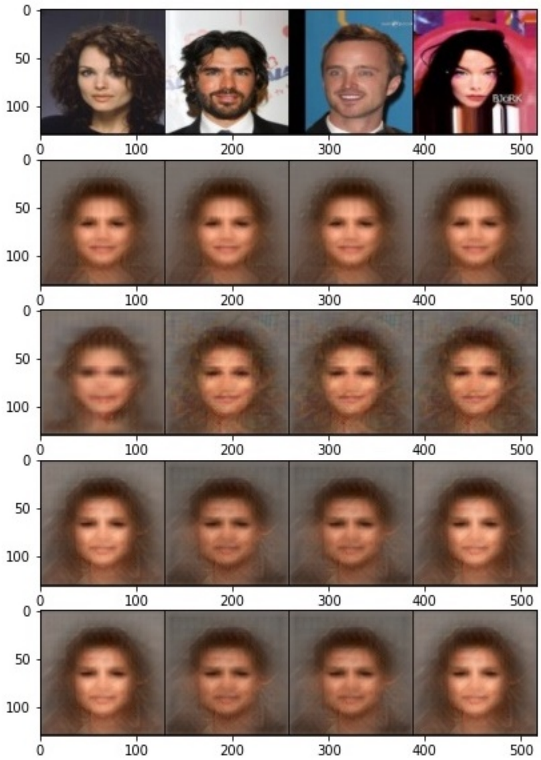


Figure 7. Visual performance of reconstructed output from the protected features for different correlation measures. Firs row: original images, row 2, 3: DistCorr for $\gamma = 2, 20$, row 4, 5: HSIC for $\gamma = 2, 20$.

5. Defense Against an Attribute Inference Attack

Herein, our primary focus is to design a framework for defense against attribute inference attacks. The defender attempts to share a representation with relevant information about the target class label, but keeps the sensitive attribute private.

The model consists of four modules: encoder, obfuscator, target classifier, and adversary classifier. The encoder is trained using supervised and private contrastive loss to provide maximal discrimination for the classification task while protecting the private attribute. Furthermore, the encoded features are obfuscated, and the target classifier is jointly trained to maintain the classification performance. Finally, adversarial training is applied between the target classifier and the adversary classifier.

5.1. Proposed Architecture

The overall private data-sharing framework, shown in Figure 8, consists of four steps:

1. An *encoder* f_ϕ is pre-trained on the public data using supervised and private contrastive loss. The encoder is later used to extract discriminative representation for the targeted classification task;
2. An *obfuscator* f_ψ is learned to remove relevant information in the representation $\mathbf{h}$ about the private attribute;
3. A *target classifier* g_{θ_t} is jointly trained with the obfuscator to ensure that the useful information for the intended classification task is preserved in the obfuscated representation;
4. An *adversary classifier* g_{θ_a} is adversely trained to minimize the classification error for the private attribute.

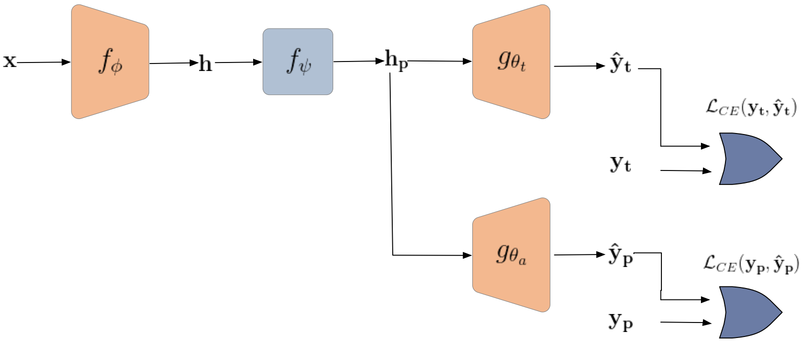

x
f_ϕ
h
f_ψ
$\mathbf{h_p}$
g_{θ_t}
$\hat{\mathbf{y}}_\mathbf{t}$
$\mathcal{L}_{CE}(\mathbf{y_t}, \hat{\mathbf{y}}_\mathbf{t})$
$\mathbf{y_t}$
g_{θ_a}
$\hat{\mathbf{y}}_\mathbf{p}$
$\mathcal{L}_{CE}(\mathbf{y_p}, \hat{\mathbf{y}}_\mathbf{p})$
$\mathbf{y_p}$

Figure 8. General diagram of the proposed framework for defense against an attribute inference attack.

5.1.1. Encoder

As displayed in Figure 3, the encoder f_ϕ is initially trained with supervised and private contrastive loss to output a well-discriminated feature representation and protect the private attributes. As mentioned in the previous section, the key idea behind contrastive loss is to push negative pairs apart and pull positive ones close. In a supervised contrastive loss, the positive pairs are those with the same target labels. Maximal discrimination can thus be achieved with respect to the target class.

This concept can be further extended to preserve the privacy of private attributes by allowing minimal discrimination regarding the sensitive label. In other words, for a supervised and private contrastive loss, we will assume:

- Positive pairs: Those with the same target label as the anchor image;
- Negative pairs: Those with the different target labels and the same private label as the anchor image.

Therefore, for an augmented dataset of $\mathcal{D} = \{(\mathbf{x}_{1,i}, \mathbf{x}_{1,j}, \mathbf{y}_{1,t}, \mathbf{y}_{1,p}), \ldots (\mathbf{x}_{N,i}, \mathbf{x}_{N,j}, \mathbf{y}_{N,t}, \mathbf{y}_{N,p})\}$, we can define the positive and negative set for each sample $\mathbf{x}_k$ as:

$$\begin{aligned} P(\mathbf{x}_k) &= \{(\mathbf{x}_{l,i}, \mathbf{x}_{l,j}) \quad \text{if} \quad (\mathbf{y}_{k,t} = \mathbf{y}_{l,t}\}_{l=1}^{N}, \\ N(\mathbf{x}_k) &= \{(\mathbf{x}_{l,i}, \mathbf{x}_{l,j}) \quad \text{if} \quad (\mathbf{y}_{k,t} \neq \mathbf{y}_{l,t} \quad \& \quad \mathbf{y}_{k,p} = \mathbf{y}_{l,p})\}_{l=1}^{N}. \end{aligned} \tag{14}$$

The supervised and private contrastive loss based on SupCon [19] can thus be defined as:

$$\mathcal{L}_{private-supcon} = -\sum_{i} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_p))}{\sum_{k \in N(i)} \exp(\text{sim}(\mathbf{z}_i, \mathbf{z}_k))}, \tag{15}$$

where $P(i)$ and $N(i)$ denote positive and negative sets with respect to sample $\mathbf{x}_i$. Similar to SupCon [19], Dai et al. introduced a supervised contrastive loss based on Momentum Contrast (MoCo) [31] denoted as UniCon [32]:

$$\mathcal{L}_{unicon} = \log\left(1 + \sum_{\{k^-\}} \exp\left(s_{k^-}\right) \sum_{\{k^+\}} \exp\left(-s_{k^+}\right)\right), \tag{16}$$

where s denotes the similarity score and $\{k^-\}$, $\{k^+\}$ are the subset of negative and positive pairs, respectively. Likewise, we can extend UniCon loss to take into account private and sensitive attributes as:

$$\mathcal{L}_{private-unicon} = \log(1 + \sum_{k^- \in N(x_k)} \exp\left(s_{k^-}\right) \sum_{k^+ \in P(x_k)} \exp\left(-s_{k^+}\right)). \tag{17}$$

5.1.2. Obfuscator

The obfuscator f_ψ is trained to hide sensitive and private attributes from the shared representation while keeping the relevant information regarding the target class label.

5.1.3. Target Classifier

The classifier g_{θ_t} is a lightweight neural network with three fully connected layers and Relu activation functions. The classifier is jointly trained with the obfuscator to maintain the classification accuracy for the target class label:

$$(\hat{\theta}_t, \hat{\psi}) = \text{argmin}_{\theta_t, \psi} \mathcal{L}_{CE}(\mathbf{y}_t, \hat{\mathbf{y}}_t), \tag{18}$$

where $\mathcal{L}_{CE}$ indicates the cross-entropy between the target attribute $\mathbf{y}_t$ and its estimate $\hat{\mathbf{y}}_t$.

5.1.4. Adversary Classifier

The adversary classifier g_{θ_a} plays the role of an attacker attempting to infer private attributes using the eavesdropped features. We simulate a game between the adversary and the defender through an adversarial training procedure. The attacker tries to minimize the classification error for the private attributes as:

$$\hat{\theta_a} = \text{argmin}_{\theta_a} \mathcal{L}_{CE}(\mathbf{y}_p, \hat{\mathbf{y}}_p). \tag{19}$$

Meanwhile,the defender aims to degrade the performance of the adversary classifier and minimize the private attribute leakage while maintaining good performance on the target classification task. Hence:

$$\hat{\psi} = \text{argmin}_{\psi} \mathcal{L}_{CE}(\mathbf{y}_t, \hat{\mathbf{y}}_t) - \gamma \mathcal{L}_{CE}(\mathbf{y}_p, \hat{\mathbf{y}}_p), \tag{20}$$

where γ is the utility-privacy trade-off parameter. Algorithm 1 delineates the overall steps in our proposed adversarial training procedure.

Algorithm 1 Adversarial Training Procedure

```
**Input**: dataset $\mathcal{D}$ and parameter $\gamma$
**Output**: $\phi, \psi, \theta_t, \theta_a$
for every epoch do
    Sample a minibatch from dataset
    Train $\phi$ using $\mathcal{L}_{private-supcon}$ or $\mathcal{L}_{private-unicon}$ in Equations (15) and (16)
end for
for every epoch do
    Sample a minibatch from dataset
    Train $\psi$ to minimize $\mathcal{L}_{CE}(\mathbf{y}_t, \hat{\mathbf{y}}_t) - \gamma\mathcal{L}_{CE}(\mathbf{y}_p, \hat{\mathbf{y}}_p)$
    Train $\theta_a$ to minimize $\mathcal{L}_{CE}(\mathbf{y}_p, \hat{\mathbf{y}}_p)$
    Train $\theta_t$ to minimize $\mathcal{L}_{CE}(\mathbf{y}_t, \hat{\mathbf{y}}_t)$
end for
```

5.2. Experimental Results

This section analyzes the effectiveness of the proposed framework. For the rest of this section, we refer to utility as the classification accuracy on the target class label. Similarly, privacy is defined as the classification performance on the private and sensitive attribute.

5.2.1. Experimental Setup

Dataset: We conducted experiments on a celebrity face image dataset, CelebA [27], which consists of over 20,000 celebrity images, where each image is annotated with 40 attributes. Every input image is center-cropped by 178 × 178 and then resized to 128 × 128. We select the "gender" attribute for our intended classification task and "age" with two classes of *young* and *old* as the sensitive attribute.

Attacker setup: The adversary has a set of publicly available protected representations $\mathbf{h}_p$ with the corresponding original images $\mathbf{x}$ and their protected labels $\mathbf{y}_p$ and aims to train a classifier to re-identify the protected attribute.

Defender setup: The primary goal of the defender is two-fold: the defender aims to preserve the high accuracy of classification expressed by "target accuracy" with respect to the utility attribute $\mathbf{y}_t$. At the same time, the defender wishes to decrease the correct classification accuracy on the attacker's side, which is represented by "private accuracy" with respect to the protected attribute $\mathbf{y}_p$. The privacy utility trade-off is controlled by different values of γ in Equation (20). This trade-off is best achieved when, firstly, the publicly available representation $\mathbf{h}_p$ is discriminative with respect to the target attribute. Secondly, there needs to be an obfuscation mechanism to remove relevant information in $\mathbf{h}_p$ regarding the private attribute.

5.2.2. Impact of the Obfuscator

In this section, we investigate the impact of the obfuscator. Therefore, keeping the encoder constant, we design an attribute inference attack to classify the private and sensitive attribute with and without the obfuscator. To analyze the privacy trade-off, we experimented with different values of γ in Equation (20), and the results are reported in Table 3.

As shown in Table 3, the classification accuracy significantly drops when the obfuscation is applied, thus validating the effectiveness of the obfuscator module. The obtained results show that the decline in utility is significantly small with only a 0.3–0.7% decrease in target accuracy. Moreover, the increase in γ decreases the private classification accuracy. However, in view of privacy protection, random guessing is the ultimate goal in a binary classification setting, as the adversary can flip his guess for any accuracy lower than the random guessing threshold. In order to account for this, the flipping accuracies are also reported in the last row of Table 3 accordingly. For the CelebA dataset, the class label "age" is slightly imbalanced and distributed as 75–25%; thereby, the corresponding random guessing threshold is 62.5% (0.75 × 0.75 + 0.25 × 0.25 = 0.625). Therefore, from a privacy protection point of view, the best result is obtained for $\gamma = 1$ for UniCon loss.

Table 3. Classification accuracy on the CelebA dataset on target and private attributes for UniCon and SupCon loss and different values of γ.

Accuracy	**UniCon**			**Supcon**			**w/o obfs.**
	$\gamma = 1$	$\gamma = 2$	$\gamma = 10$	$\gamma = 1$	$\gamma = 2$	$\gamma = 10$	
target accuracy	98.37	98.34	98.30	98.33	98.31	98.30	98.34
private accuracy	32.5	25.32	19.58	25.32	25.32	17.89	82.2
100 − private accuracy	67.5	74.68	80.42	74.68	74.68	82.11	-

5.2.3. Privacy-Utility Trade-Off Comparison

To better evaluate the effectiveness of the proposed framework model, the privacy–utility trade-off for different scenarios has been investigated. The results in Table 3 validate the effectiveness of the obfuscator module. Putting the obfuscator aside, we are interested in analyzing the impact of using supervised and private loss compared to the conventional contrastive loss in Equation (2). To evaluate that, we considered the following scenarios:

- $\mathbf{h}$: the feature representations of original images from an encoder trained with a conventional contrastive loss in Equation (2);
- $\mathbf{h}_{x_{\text{noisy}}}$: the feature representations of noisy images from an encoder trained with a conventional contrastive loss in Equation (2);
- $\mathbf{h}_{\text{noisy}}$: noisy feature representations of original images from an encoder trained with a conventional contrastive loss in Equation (2);
- $\mathbf{h}_{\text{private}-\text{unicon}}$: the feature representations of original images from an encoder trained with private UniCon loss in Equation (16);
- $\mathbf{h}_{\text{private}-\text{supcon}}$: the feature representations of original images from an encoder trained with private SupCon loss in Equation (15);
- $\mathbf{h}_{p_{\text{unicon}}}$: the obfuscated and protected features of the proposed framework using UniCon loss in Equation (16);
- $\mathbf{h}_{p_{\text{supcon}}}$: the obfuscated and protected features of the proposed framework using SupCon loss in Equation (15).

The privacy–utility tradeoff in the form of target and private accuracy for various settings is reported in Table 4. The final accuracies were flipped in cases lower than the random guessing threshold for a fair comparison.

Impact of supervised and private contrastive loss: As reported in Table 4, the accuracy on the target class is higher for both cases of SupCon and Unicon compared to the unsupervised contrastive loss. This is mainly due to the fact that there was no label information used in the conventional contrastive loss (Equation (2)). In addition, the accuracy on the private attribute is 4% lower in $\mathbf{h}_{\text{private}-\text{supcon}}$ and $\mathbf{h}_{\text{private}-\text{supcon}}$ compared to $\mathbf{h}$, showing benefits of using supervised and private loss.

Impact of adding noise: Adding noise to raw images or extracted features can be considered as a defense mechanism. Injecting Gaussian noise into the data has been widely used in federated learning [33,34]. Indeed, the results in Table 4 demonstrate that the privacy increases as we add noise to the images or the features. Moreover, raising the variance of the noise leads to more privacy gain. However, the private classification accuracies for noisy data are still far from the results we can achieve using the proposed framework. Besides, by adding noise, we also lose utility as the target accuracy drops.

Comparison to DeepObfuscator [16]: We carefully explored and examined other papers in state-of-the-art for a fair comparison. Unfortunately, the differences in the problem formulation make this comparison difficult and unfair in some cases. For example, several works have studied the privacy leakage of a face verification system different from the attribute classification problem formulation. In ref. [35], the authors proposed an adversarial framework for reducing gender information in the final embedding vectors used for the verification system. Hence, we can argue that even though the privacy task of attribute leakage in the embeddings is the same, the utility is defined differently, thereby making the comparison infeasible.

Moreover, several studies have investigated the same utility–privacy formulation as our proposed framework. However, they differ in their overall setting. For example, Boutet et al. [36] proposed a privacy-preserving framework against attribute inference attacks in a federated learning setting. In their experiments, the main target label is "smiling," while the protected label is the "gender" of users.

Nevertheless, a very similar problem formulation and setting are studied in ref. [16]. Li et al. [16] exploit an adversarial game to maintain the classification performance on the public class label while preserving against an attribute-inference attack. As they have used different attributes as the target and private, we re-run their obfuscator model for our public and private attributes. The DeepObfuscator model in [16] is further adapted to only consider the attribute inference attack. The results reported in Table 4 demonstrate the superior performance of the proposed method compared to DeepObfucator.

Table 4. Privacy-Utility trade-off.

	Accuracy	
	Target Accuracy	**Private Accuracy**
$\mathbf{h}$	98.24	86.3
$\mathbf{h}_{x_{\text{noisy}}}$	98.03	86.05
$\mathbf{h}_{\text{noisy}}$	97.61	84.5
$\mathbf{h}_{\text{private-unicon}}$	98.38	82.23
$\mathbf{h}_{\text{private-supcon}}$	98.34	82.2
our: $\mathbf{h}_{p_{\text{unicon}}}$ and $\gamma = 1$	98.30	**67.5**
our: $\mathbf{h}_{p_{\text{supcon}}}$ and $\gamma = 1$	98.30	**74.68**
DeepObfuscator [16]	97.75	76.03

6. Conclusions

This paper addressed the problem of template protection against the most commonly used attacks, namely, reconstruction and attribute inference attacks. Two defense frameworks based on contrastive learning were proposed.

For defense against the reconstruction attack, we directly minimize the correlation and dependencies of encoded features with the original data, avoiding the unnecessary complications of a min-max adversarial training. Furthermore, training an encoder with the supervised contrastive loss would minimize discrimination in the feature space and remove redundant information about the original images. Hence, there is no substantial loss in classification performance, and the proposed framework provides a better utility-privacy trade-off.

In the attribute inference attack, the adversary wishes to access the private attribute given the shared protected templates. Therefore, in the first defense step, we propose an encoder trained with the supervised and private contrastive loss. Furthermore, an obfuscator module is trained in an adversarial manner to preserve the privacy of private attributes while maintaining a good classification performance. The reported results on the CelebA dataset validate the effectiveness of the proposed framework. The future work aims at designing a framework based on contrastive loss considering both reconstruction and attribute inference attacks. Another interesting avenue of research is to investigate the performance of the proposed framework on other datasets.

Author Contributions: Conceptualization, S.R. and S.V.; methodology, S.R. and V.K.; data curation, M.A.J.; supervision, S.V.; investigation, S.R.; validation, S.R. and V.K. and M.A.J.; visualization, S.R. and M.A.J.; writing—original draft, S.R.; Writing—review & editing, S.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html (accessed on 2 January 2022).

Conflicts of Interest: The authors declare no conflict of interest.

Reference

1. Tanuwidjaja, H.C.; Choi, R.; Baek, S.; Kim, K. Privacy-Preserving Deep Learning on Machine Learning as a Service—A Comprehensive Survey. *IEEE Access* **2020**, *8*, 167425–167447. [CrossRef]
2. Mai, G.; Cao, K.; Yuen, P.C.; Jain, A.K. On the Reconstruction of Face Images from Deep Face Templates. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1188–1202. [CrossRef] [PubMed]
3. Hill, S.; Zhou, Z.; Saul, L.; Shacham, H. On the (in) effectiveness of mosaicing and blurring as tools for document redaction. *Proc. Priv. Enhancing Technol.* **2016**, *2016*, 403–417. [CrossRef]
4. Boracchi, G.; Foi, A. Modeling the performance of image restoration from motion blur. *IEEE Trans. Image Process.* **2012**, *21*, 3502–3517. [CrossRef] [PubMed]
5. Vishwamitra, N.; Knijnenburg, B.; Hu, H.; Kelly Caine, Y.P. Blur vs. block: Investigating the effectiveness of privacy-enhancing obfuscation for images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 39–47.
6. Tekli, J.; Al Bouna, B.; Couturier, R.; Tekli, G.; Al Zein, Z.; Kamradt, M. A Framework for Evaluating Image Obfuscation under Deep Learning-Assisted Privacy Attacks. In Proceedings of the 2019 17th International Conference on Privacy, Security and Trust (PST), Fredericton, NB, Canada, 26–28 August 2019; pp. 1–10.
7. McPherson, R.; Shokri, R.; Shmatikov, V. Defeating image obfuscation with deep learning. *arXiv* **2016**, arXiv:1609.00408.
8. Li, T.; Lin, L. AnonymousNet: Natural Face De-Identification With Measurable Privacy. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 56–65.
9. Ren, Z.; Lee, Y.J.; Ryoo, M.S. Learning to anonymize faces for privacy preserving action detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 620–636.
10. Li, J.; Han, L.; Chen, R.; Zhang, H.; Han, B.; Wang, L.; Cao, X. Identity-Preserving Face Anonymization via Adaptively Facial Attributes Obfuscation. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021; pp. 3891–3899.
11. Ogburn, M.; Turner, C.; Dahal, P. Homomorphic encryption. *Procedia Comput. Sci.* **2013**, *20*, 502–509. [CrossRef]
12. Ossia, S.A.; Taheri, A.; Shamsabadi, A.S.; Katevas, K.; Haddadi, H.; Rabiee, H.R. Deep Private-Feature Extraction. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 54–66. [CrossRef]
13. Li, M.; Lai, L.; Suda, N.; Chandra, V.; Pan, D.Z. Privynet: A flexible framework for privacy-preserving deep neural network training. *arXiv* **2017**, arXiv:1709.06161.
14. Pittaluga, F.; Koppal, S.; Chakrabarti, A. Learning privacy preserving encodings through adversarial training. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 26–28 August 2019; pp. 791–799.
15. Liu, S.; Du, J.; Shrivastava, A.; Zhong, L. Privacy adversarial network: representation learning for mobile data privacy. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*; Association for Computing Machinery: New York, NY, USA, 2019; Volume 3, pp. 1–18.
16. Li, A.; Guo, J.; Yang, H.; Salim, F.D.; Chen, Y. DeepObfuscator: Obfuscating Intermediate Representations with Privacy-Preserving Adversarial Learning on Smartphones. *arXiv* **2019**, arXiv:1909.04126.
17. Huang, C.; Kairouz, P.; Chen, X.; Sankar, L.; Rajagopal, R. Context-aware generative adversarial privacy. *Entropy* **2017**, *19*, 656. [CrossRef]
18. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*; PMLR: London, UK, 2020; pp. 1597–1607.
19. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised contrastive learning. *arXiv* **2020**, arXiv:2004.11362.
20. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
21. Gretton, A.; Bousquet, O.; Smola, A.; Schölkopf, B. Measuring statistical dependence with Hilbert-Schmidt norms. In *International Conference on Algorithmic Learning Theory*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 63–77.
22. Gretton, A.; Fukumizu, K.; Teo, C.H.; Song, L.; Schölkopf, B.; Smola, A.J. A kernel statistical test of independence. In Proceedings of the 20th International Conference on Neural Information Processing Systems (NIPS 2007), Vancouver British, BC, Canada, 3–6 December 2007; pp. 585–592.
23. Székely, G.J.; Rizzo, M.L.; Bakirov, N.K. Measuring and testing dependence by correlation of distances. *Ann. Stat.* **2007**, *35*, 2769–2794. [CrossRef]
24. Sun, J.; Yao, Y.; Gao, W.; Xie, J.; Wang, C. Defending against Reconstruction Attack in Vertical Federated Learning. *arXiv* **2021**, arXiv:2107.09898.

25. Kasieczka, G.; Shih, D. Robust Jet Classifiers through Distance Correlation. *Phys. Rev. Lett.* **2020**, *125*, 122001. [CrossRef]
26. Vepakomma, P.; Singh, A.; Gupta, O.; Raskar, R. NoPeek: Information leakage reduction to share activations in distributed deep learning. *arXiv* **2020**, arXiv:cs.LG/2008.09161.
27. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep Learning Face Attributes in the Wild. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3730–3738.
28. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
29. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
30. Wang, Z.; Simoncelli, E.P.; Bovik, A.C. Multiscale structural similarity for image quality assessment. In Proceedings of the Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, Pacific Grove, CA, USA, 9–12 November 2003; Volume 2, pp. 1398–1402.
31. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual Event, 13–19 June 2020; pp. 9729–9738.
32. Dai, Z.; Cai, B.; Lin, Y.; Chen, J. UniMoCo: Unsupervised, Semi-Supervised and Full-Supervised Visual Representation Learning. *arXiv* **2021**, arXiv:2103.10773.
33. Papernot, N.; Song, S.; Mironov, I.; Raghunathan, A.; Talwar, K.; Erlingsson, Ú. Scalable private learning with pate. *arXiv* **2018**, arXiv:1802.08908.
34. Truex, S.; Baracaldo, N.; Anwar, A.; Steinke, T.; Ludwig, H.; Zhang, R.; Zhou, Y. A hybrid approach to privacy-preserving federated learning. In Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security, London, UK, 15 November 2019; pp. 1–11.
35. Dhar, P.; Gleason, J.; Souri, H.; Castillo, C.D.; Chellappa, R. Towards gender-neutral face descriptors for mitigating bias in face recognition. *arXiv* **2020**, arXiv:2006.07845.
36. Boutet, A.; Lebrun, T.; Aalmoes, J.; Baud, A. MixNN: Protection of Federated Learning against Inference Attacks by Mixing Neural Network Layers. *arXiv* **2021**, arXiv:2109.12550.