A PRIVACY-PRESERVING IMAGE CLASSIFICATION FRAMEWORK WITH A LEARNABLE OBFUSCATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Real world images often contain large amounts of private / sensitive information that should be carefully protected without reducing their utilities. In this paper, we propose a privacy-preserving deep learning framework with a learnable obfuscator for the image classification task. Our framework consists of three models: learnable obfuscator, classifier and reconstructor. The learnable obfuscator is used to remove the sensitive information in the images and extract the feature maps from them. The reconstructor plays the role as an attacker, which tries to recover the image from the feature maps extracted by the obfuscator. In order to best protect users' privacy in images, we design an adversarial training methodology for our framework to optimize the obfuscator. Through extensive evaluations on real world datasets, both the numerical metrics and the visualization results demonstrate that our framework is qualified to protect users' privacy and achieve a relatively high accuracy on the image classification task.

1 INTRODUCTION

In the past few years, deep neural networks (DNNs) (Goodfellow et al., 2016) have achieved great breakthroughs in computer vision, speech recognition and many other areas. To support the training of DNNs, large datasets have been collected, e.g., ImageNet (Deng et al., 2009), MNIST (LeCun et al., 1998) and CIFAR-10/CIFAR-100 (Krizhevsky & Hinton, 2009) as image datasets, Youtube-8M (Abu-El-Haija et al., 2016) as video datasets, and AudioSet (Gemmeke et al., 2017) as audio datasets. These datasets are usually crowdsourced from the real world, and may carry sensitive private information, thus, leading to serious *privacy problems*.

The new European Union's General Data Protection Regulation (GDPR) (Regulation, 2016) stipulates that personal data cannot be stored for long periods of time, and personal data requests, such as deleting personal images, should be handled within 30 days. In other words, this regulation prevents long-term storage of video/image data (e.g., from CCTV cameras), which hinders the collection of real-world datasets for training deep learning models. However, the data storage limitations do not apply if the data is anonymized.

This regulation considers the trade-off between the utility and the privacy of the data. However, 30 days may not be a long enough period to collect image data and train a complex deep learning model, and deletion of data hinders re-training later when the model structure is updated or more data becomes available. GPDR allows anonymized data to be stored indefinitely, which inspires us to design a framework where an image is converted into an obfuscated intermediate representation that removes sensitive personal information while retaining suitable discriminative features for the learning task. Thus the obfuscated intermediate representation can be stored indefinitely for model training in compliance with GDPR.

Contributions In this paper, we design a obfuscator-adversary framework to obtain a trainable obfuscator that fulfills the dual goals of removing sensitive information and extracting useful features for the learning task. Here, we mainly focus on image classification as the learning task, since it is a more general task in computer vision – the framework could be extended to other tasks. Our framework consists of three models, each with its own objective: the obfuscator, the classifier and the reconstructor, shown in Figure 1. The *obfuscator* works as an information remover, which takes the input image and extracts feature maps that carry enough primary information for the classification task while removing sensitive private information. These feature maps are the obfuscated representation of the input image. The *classifier* uses the obfuscated representation to perform classification of the input image. Finally, the *reconstructor* plays the role as an adversary whose goal is to extract the sensitive information from the obfuscated representation.

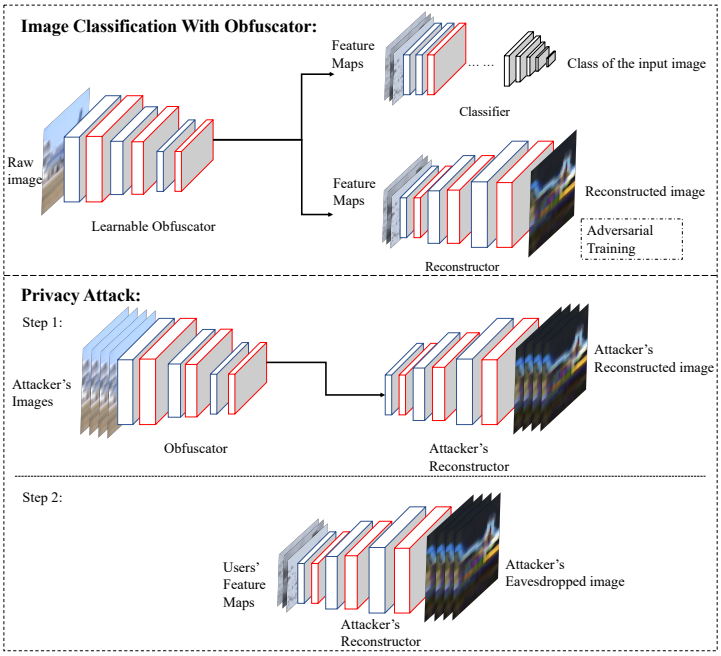


Figure 1: (top) Our proposed framework learns an obfuscated representation (feature maps) for image classification that also prevents leakage of users' privacy. The *obfuscator* extracts a feature map (the obfuscated representation) that both prevents the reconstruction of the image and keeps the primary information for the classification task. The *classifier* uses the obfuscated feature map to perform the image classification task. The *reconstructor* aims to reconstruct the original image from the feature map. The three models are trained using an adversarial training process. (bottom) The attacker aims to reconstruct a users' images to eavesdrop their privacy. We assume that the attacker has unlimited access to the obfuscator and the feature maps extracted from users' images. The attacker trains their own reconstructor using their own set of images, and attempts to reconstruct the users' images from the stored feature maps.

As different kinds of images may contain different kinds of sensitive information (e.g., personal identity, location, etc), we choose image reconstruction quality as a general measure for privacy preservation. The reconstructor, as the adversary, tries to reveal the sensitive information by restoring the image from the feature maps. If even state-of-the-art reconstructors cannot restore the image, and the classification accuracy is still good, we can say that our framework has experimentally demonstrated enough security to protect users' privacy. As the obfuscator and the reconstructor have opposite objectives, the training of our proposed framework can be formalized as an adversarial training paradigm. The main contributions of this paper are:

1) To the best of our knowledge, this is the first study of using the adversarial training methodology for privacy-preserving image classification.
2) We propose a brute-force experimental evaluation method to demonstrate the security-level performance of the proposed framework.
3) The experiments on real-world datasets demonstrate that utility(classification accuracy)-privacy trade-off is perfectly handled via the adversarial training process.

2 Related Work

Deep learning requires a tremendous amount of data that may contain a significnat private information. Conventional works have already proposed several approaches to counter the privacy problem in learning tasks. Prior works can be divided into three categories: privacy of datasets, privacy of models, and privacy of models' outputs (Shokri & Shmatikov, 2015). In this paper, we mainly focus on the privacy of datasets.

One way to protect the privacy of data is to increase the amount of uncertainty, e.g., based on k-anonymity (Sweeney, 2002), l-diversity (Machanavajjhala et al., 2006) and t-closeness (Li et al.,

2007). Unfortunately, these approaches are only suitable for low-dimensional data because the quasi-identifiers and sensitive attributes are not easily defined for high-dimensional data. This makes private information in multimedia (videos, images and audios, etc.) much harder to be protected.

Differential privacy (Dwork, 2008), as the state-of-the-art privacy preserving mechanism, is a more formal way to open-source a database while keeping all individual records private by adding well-designed noise. However, differential privacy only affects inserting and deleting an individual data record. Abadi et al. (2016) investigated the application of differential privacy to deep learning, and extended the conventional Stochastic Gradient Descent (SGD) (Bottou, 2010) algorithm to a novel Differentially Private SGD (DPSGD) algorithm. However, the inherent character of differential privacy implies that there will always be a data utility and privacy tradeoff. The fact that more strict privacy guarantee always demands more noise added to the data often limits its application scenarios, especially when high accuracy of learning tasks is a must.

Another way for data-level privacy protection is to use cryptographic operations to encrypt the dataset. Gilad-Bachrach et al. (2016) proposed Cryptonets, a cloud based framework, in which the inference stage is applied on encrypted datum. However, Cryptonets has some limitations. First, it has a sensitive privacy-utility trade-off. Second, low-degree polynomials using homomorphic encryption are not able to compute the non-linear activation function efficiently. Focusing on these shortcomings, Rouhani et al. (2017) proposed DeepSecure, a provably-secure framework for scalable deep learning based data analysis. DeepSecure is also a cloud-client based framework, and it does not have the concern of privacy-utility trade-off. However, this approach is only suitable for scenarios in which the number of samples submitted by each client is less than 2600, which extremely limits its application. Other applications of homomorphic encryption to privacy preserving tasks, e.g., Chabanne et al. (2017), Bellafqira et al. (2018) and Liu et al. (2018), have almost the same disadvantages and limitations as approaches mentioned above.

Recent works extend the common deep neural networks to protect the dataset privacy using pure machine learning techniques. Osia et al. (2017; 2018) proposed a client-server model, which separates the common CNN into two parts and the first part becomes the feature extractor and the second part works as the classifier. A Siamese network is used to ensure privacy protection. However, this framework can only be deployed during the inference stage because the training of a neural network would require a large amount of communication throughput between the clients and servers. Li et al. (2017) uses the reconstruction quality as a measure for privacy preservation. However, the reconstruction quality is only used for evaluation, and it not used in the loss function during training, which makes this work similar to that of Osia et al. (2017; 2018).

In contrast to these previous works, we propose a privacy protection framework based on an adversarial training procedure, where the obfuscator and classifier work together to preserve privacy while performing the classification task, and an adversarial reconstructor tries to reveal the private information by recovering the image. As good reconstruction quality is highly related with the recovery of private information, in our framework, we include the reconstruction quality into the loss of the framework in order to better learn the obfuscator. Experimental results demonstrate that our framework both preserves privacy well and achieves good classification accuracy.

3 Security Analysis

We consider a DNN-based training service involving three entities, namely the training data providers, who provide raw images that may contain sensitive personal information and seek for protection of privacy during the process of data delivery; the service provider, who initially has all the sensitive personal information from raw images and is compelled to remove sensitive data via an obfuscator under the regulation of GDPR, while maintaining most of other useful information needed for DNN training; an attacker, who could be an unauthorized internal staff or an external hacker that intends to recover the original images from information filtered by the obfuscator.

We then focus on the strength of security against an attacker aspiring to snoop private information of the training data providers. Specifically, in our model, we consider a Chosen Image Attack (CIA), in which an attacker gains unlimited access to the obfuscator, and leverages it to generate data for training the attacker's reconstructor. We stress CIA in this paper since this is the most natural and convenient way of launching an attack. We are aware of that other data stored on the servers of the service provider, i.e., weight vectors along with training inputs and outputs of the classifier and

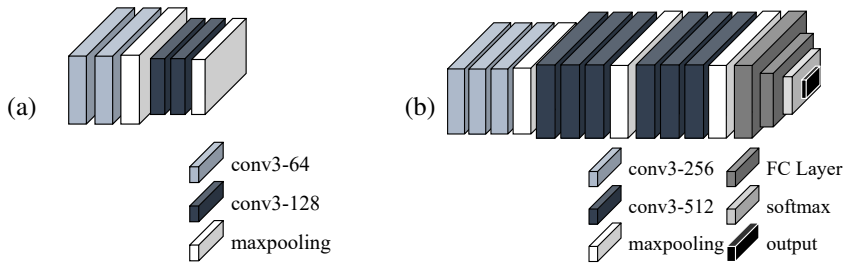


Figure 2: The structure of the (a) obfuscator and (b) classifier.

the reconstructor, can somehow also be leveraged by an attacker. However, compared to the attack domain of CIA, threats of other attacks that focus solely on the training data domain appear less imminent and hence those attacks are not covered in this paper. We further assume within the attack model of CIA, an attacker would subsequently use a DNN to carry out image reconstruction due to the fact that the obfuscated feature maps may contain information unnoticed by humans.

For quality assessment of the reconstructed images in CIA, it is extremely difficult to directly define how successful an attack can be given the specific image because sensitive information contained within different images may vary from case to case. To this end, we adopt the index measuring the quality of reconstruction, i.e., Peak Signal to Noise Ratio (PSNR), to roughly evaluate the strength of security against CIA in our design rather than define our own privacy measurement index. We say the lower the PSNR value, the more secure it indicates that our design is able to defend against such attack model of CIA. The PSNR between two images I_o and I_r with dimensions $m \times n$ is

$$\mathrm{PSNR}(I_o, I_r) = 10 \log_{10} \frac{p_{\max}^2}{\mathrm{MSE}(I_o, I_r)} = 10 \log_{10} \frac{p_{\max}^2}{\frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (I_o(i,j) - I_r(i,j))^2}, \quad (1)$$

where $p_{\max}$ is the maximum range of pixel values in an image (typically 255 for 8-bit images). Higher values of PSNR indicate that the two images are more similar.

4 Obfuscator-Adversary Framework

In this section, we introduce the our proposed framework deep learning based privacy-preserving image classification. Our approach is divided into three modules: the obfuscator, the classifier and the reconstructor. The goal of the obfuscator is to produce a feature map that removes sensitive information from the image, while also preserving primary information for the classifier. On the opposite, the reconstructor acts as an attacker that aims to reconstruct the original image from the feature map. We formulate the training of our proposed framework as an adversarial training process.

Here we give the following notations. Let $D = \{I_i, \dots I_N\}$ denote the images in our dataset, where N is the number of images, and $Y = \{y_1, \dots, y_N\}$ are the corresponding class labels, where the set of possible classes is $\mathcal{Y} = \{1, \dots, M\}$. An obfuscator $f(\cdot; \theta_f) : \mathcal{I} \to \mathcal{F}$ is a function mapping from images $\mathcal{I}$ to feature maps $\mathcal{F}$. θ_f are the parameters (weights) of the obfuscator. The classifier $g(\cdot; \theta_g) : \mathcal{F} \to \mathcal{Y}$ represents a mapping from feature maps $\mathcal{F}$ to class labels $\mathcal{Y}$. The reconstructor $r(\cdot; \theta_r) : \mathcal{F} \to \mathcal{I}$ is a function mapping from feature maps back to images.

4.1 Obfuscator and Classifier

Conventional deep learning models for the image classification are usually based on convolutional neural networks (CNNs), which is a stack of multiple convolutional layers, pooling layers, activation functions and fully connected layers. An intrinsic characteristic of the convolutional layers in CNNs is the ability to extract discriminative information from the input image into feature maps, while ignoring non-discriminative information. This phenomenon inspires us to modify the objective of the convolutional layers to both extract discriminative features and remove sensitive information in the extracted feature maps. Thus, in our framework, we divide a deep CNN architecture, VGG16 (Simonyan & Zisserman, 2014), into two parts. The first part is used as the obfuscator, while the second part is used as the classifier. The feature map between the two parts is the obfuscated representation. Figure 2 shows the structure of the obfuscator and the classifier.

As an feature extractor and sensitive information filter, the obfuscator has two objectives. First, it should minimize the classification error to ensure the high utility of our framework. Second, to protect the privacy in input images, it should minimize the PSNR between the original image and the reconstructed image in (1). The goal of the classifier is to minimize the classification error, which is consistent with the obfuscator. Hence, the obfuscator and classifier can be trained by minimizing the loss function,

$$\theta_f^*, \theta_g^* = \arg\min_{\theta_f, \theta_g} \sum_{i=1}^{N} \mathcal{L}_{cross}(y_i, g(f(I_i))) + \lambda \text{PSNR}(I_i, r(f(I_i))), \tag{2}$$

where λ is a trade-off parameter. The first term is the categorical cross-entropy loss between the ground-truth class label and the classifier prediction. The second term is the reconstruction loss, based on the output of the reconstructor. The loss in Eq.2 is the adversarial loss of our framework.

4.2 RECONSTRUCTOR

The reconstructor in our framework plays the role of an attacker. According to the assumptions in Section 3, the attacker can access the feature maps of raw images extracted by a pre-trained obfuscator. The attacker's goal is to recover private information stored in the feature maps through reconstruction of the image from the feature map. Consequently, the objective of the reconstructor is to maximize the similarity between reconstructed images and raw images, as measured by PSNR,

$$\theta_r^* = \arg\max_{\theta_r} \sum_{i=1}^{N} \text{PSNR}(I_i, r(f(I_i))), \tag{3}$$

which is the opposite objective of the loss function in (2). The architecture of the reconstructors is discussed in the next section.

4.3 ADVERSARIAL TRAINING METHODOLOGY

Intuitively, the roles of the obfuscator and the reconstructor are working against each other. During the training period, the obfuscator tries its best to maximally remove the sensitive information in the input images so that the reconstructor cannot reconstruct images similar to raw input images. Whereas the reconstructor will fine-tune its parameters to find the best reconstructions for given input feature maps. This training formulation is exactly an adversarial training process, where two models play a minimax game (Chen et al., 2016). The adversarial training methodology introduces an rebuttal procedure in the training process. We formalize the training procedure in Algorithm 1.

Data: The training set of images $D = \{I_1, ..., I_N\}$ and their class labels $Y = \{y_1, ..., y_n\}$, number of main iterations T_{main}, number of sub-iterations T_{sub}
Result: Weights of three models: $\theta_f, \theta_g, \theta_r$ such that the given three objects are optimized
Initialization: Initialize $\theta_f, \theta_g, \theta_r$ using Xaiver initialization (Glorot & Bengio, 2010);
while *not converged or reached* T_{main} **do**
  Generate augmented data from input images;
  if *is the first epoch* **then**
    train the obfuscator-classifier until it reaches its optimal performance (at least 200 epochs);
  else
    train the obfuscator-classifier for T_{sub} epochs;
  end
  freeze the obfuscator, then train the reconstructor for T_{sub} epochs;
  freeze the reconstructor and classifier, then train the obfuscator for T_{sub} epochs;
end

Algorithm 1: Adversarial training algorithm for our framework

As the primary task of our framework is to classify images, we first train the classifier and obfuscator together without any security concern to obtain optimal performance at image classification. The obfuscator working in this stage can be recognized as the first several layers of the classifier.

Table 1: Reconstructor Configurations

SimRec	**URec#1**	**URec#2**	**ResRec**
Input ($8 \times 8 \times 128$ feature maps)			
conv3-128 conv3-128	conv3-64 conv3-64 conv3-64	conv3-64 conv3-64	transconv3-64 $\begin{bmatrix} 3 \times 3, & 64 \\ 3 \times 3, & 64 \end{bmatrix} \times 2$
Upsampling2D	Upsampling2D	Upsampling2D	Upsampling2D
conv3-64 conv3-64	conv3-128 conv3-128 conv3-128	conv3-128 conv3-128	$\begin{bmatrix} 3 \times 3, & 64 \\ 3 \times 3, & 64 \end{bmatrix} \times 2$
Upsampling2D	Upsampling2D	Upsampling2D	Upsampling2D
conv3-3	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256	conv1-3
sigmoid	conv3-3 conv1-3	conv3-3 conv1-3	sigmoid
	sigmoid	sigmoid	

After the initial training of the obfuscator-classifier, we need to take the privacy problem into consideration, which is handled by our adversarial training framework. In particular, the reconstructor is trained while holding the obfuscator fixed, and then the obfuscator is trained while keeping the reconstructor and classifier fixed. In this way, the obfuscator can counteract the any improvements in PSNR from the reconstructor. Finally, the whole procedure is repeated until convergence or the maximum number of epochs is reached.

5 Experiments

In our experiments, we use three datasets: MNIST handwritten digits dataset (LeCun et al., 1998) and CIFAR-10/CIFAR-100 dataset (Krizhevsky & Hinton, 2009). MNIST consists of 70,000 handwritten digit images, of which 60,000 images belong to the training set, and 10,000 images belong to the testing set. All these images are size-normalized and centered to a fixed-size (28×28). CIFAR-10 is a tiny image dataset containing 60,000 32×32 colored images in 10 classes (50,000 for training and 10,000 for testing). CIFAR-100 is similar to CIFAR-10 but contains 100 classes with 600 images per class.

In order to ensure the security under different kinds of attackers, we implement 4 state-of-the-art reconstructors as attackers, and train them separately within our framework. Note that for space interest, we only report the results where we set trade-off parameter $\lambda = 1$ in the experiment. We will include more enriched results with diverse choice of parameters in the full version. The four reconstructors are (see Table 1 for architecture details): 1) Simple autoencoder reconstructor (SimRec): This is a simple reconstructor, which is just a reversed model of the obfuscator used in our framework. As the structure of the obfuscator-reconstructor is similar to the structure of an autoencoder; 2) U-net reconstructor #1 (URec#1): U-net (Ronneberger et al., 2015) is a fully convolutional neural network structure used for biomedical image segmentation and image generation with GANs Isola et al. (2016); 2) U-net reconstructor #2 (URec#2): This reconstructor is a simpler version of URec#1, which reduces the number of layers; 3) ResNet reconstructor (ResRec): ResNet (He et al., 2016) is a deep learning model used for image recognition, as well as image restoration (Jiao et al., 2017). ResNet model involves the residual function and contains several residual blocks.

5.1 Classification Accuracy and Reconstruction Results

In this section, we will show the experimental results to demonstrate that our framework is a strong method to protect user's privacy while keeping the utility of a deep learning image classification model. Table 2 presents the classification accuracy for each dataset, related to the number of epochs. As the space is limited, here we only give the accuracy of the first 100 epochs. During the training and testing process, we find that the accuracy is not highly related to the reconstructor, which means that although we changed the reconstructor in our framework, the classification accuracy is consistent (as shown in Figure 3). This suggests that the obfuscator is robust to the type of the reconstructor, so that our framework is able to be deployed in different scenarios. In Table 2, the accuracy

Table 2: Average classification accuracy versus epoch. Epoch 1 corresponds to the baseline classifier.

#epoch/dataset	1 (baseline)	10	20	30	40	50	60	70	80	90	100
MNIST	99.70%	85.17%	88.19%	90.43%	90.60%	90.53%	91.03%	91.76%	92.17%	92.84%	**92.95%**
CIFAR-10	93.56%	83.35%	84.17%	84.74%	85.33%	85.48%	87.30%	88.07%	88.37%	89.21%	**89.48%**
CIFAR-100	70.40%	50.23%	56.9%	58.25%	59.57%	60.51%	62.63%	63.18%	**64.53%**	64.21%	64.45%

(a) MNIST (b) CIFAR-10 (c) CIFAR-100

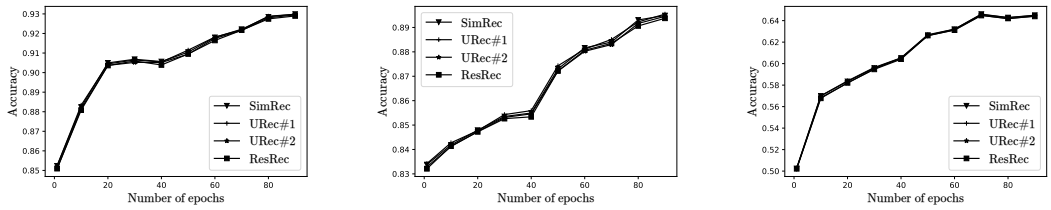


Figure 3: The classification accuracy for different datasets and different training reconstructors.

of the first epoch represents the baseline of VGG16 on the given dataset. The baseline for MNIST is 99.70% and our accuracy using privacy-preservation is 92.95%. For CIFAR-10 it is 93.56% and 89.48%, and for CIFAR-100 it is 70.40% and 64.53%. For comparison, the state-of-the-art work in privacy-preserving networks (Li et al., 2017) achieved only 70.1% average accuracy on CIFAR-10 dataset, which our framework outperforms. The major difference between our work and Li et al. (2017) is that we employ adversarial training where the goal is to both reduce reconstruction quality and improve classification accuracy.

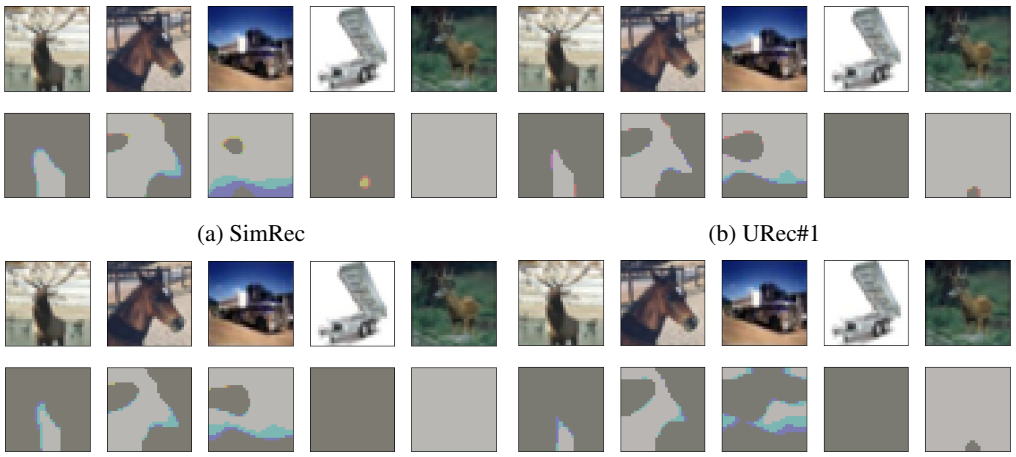
Figure 4 gives some examples of the reconstruction results on CIFAR-10 using different reconstructors. Comparing the reconstructed images with raw images, we find that the reconstructed images only contain the blurred outlines of target objects in raw images, which may represent the category information of the image. The average PSNR values for the reconstructors on all datasets is shown in the diagonal of Table 3 – SimRec achieved 28.0306 as its average PSNR, while URec#1, URec#2 and ResRec, the PSNR values are 28.0020, 28.0129 and 28.0176, respectively. Small values of PSNR indicate that the difference between the input image and the reconstructed image is large, and hence most of the information of the raw image is removed by the obfuscator.

In order to simulate the behavior of the attackers, besides the adversarial training and testing, we designed a complementary experiment that simulates a brute-force attack. In this experiment, we assume that the attacker has a pre-trained obfuscator, and then trains multiple reconstructors to try to recover sensitive information from a given feature map. During the training process, the obfuscator is not modified and only the reconstructors' weights are updated (simulating the situation that attackers wants to break the obfuscator). The off-diagonal entries of Table 3 show the reconstruction PSNR when the attacker uses a different reconstruction method than the one used for training, The PSNR values are also low and comparable to the reconstructor used for training the obfuscator. This suggests that the obfuscator is able to remove most of the sensitive information from the feature map, and that it is robust against different types of attackers on which it was not trained.

Finally, to demonstrate that adversarial training is useful, we train a variant of our framework that removes the reconstructor and the adversarial training process, and the PSNR loss term in Eq.2 is also removed. Reconstruction results are shown in Figure 5. Compared to our method, the reconstructed images without the adversarial training have more detailed information about raw images,

Table 3: Average PSNR for different training and attacking reconstructors.

		Training reconstructor			
		SimRec	URec#1	URec#2	ResRec
Attack Reconstructor	SimRec	8.0306	7.9876	7.9278	8.0058
	URec#1	8.0273	8.0020	8.9690	7.9975
	URec#2	8.0222	7.9791	8.0129	8.0090
	ResRec	8.0317	7.9925	7.9630	8.0176



(a) SimRec (b) URec#1

(c) URec#2 (d) ResRec

Figure 4: Reconstruction results using different reconstructors. For each vertical pair, the top image is the input image, and the bottom is the reconstruction

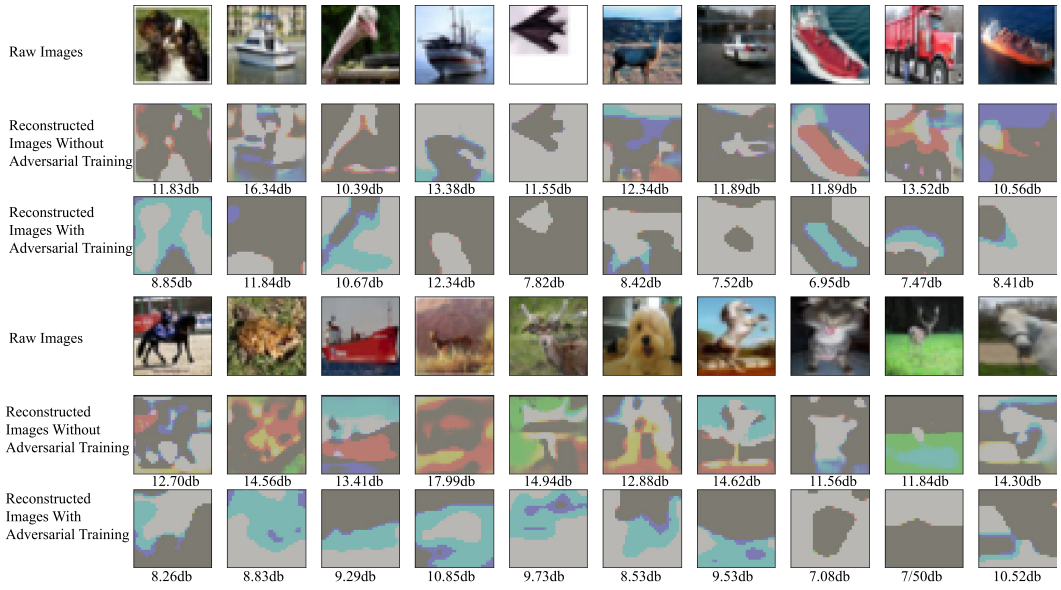


Figure 5: The comparison between reconstructed images with adversarial training and without adversarial training. The number under each image is the PSNR for the reconstruction.

and thus allow more private information to be leaked. Thus adversarial training plays an important role in learning the obfuscator.

6 Discussion and Conclusion

We proposed a deep learning framework on privacy-preserving image classification tasks. Our framework has three modules, the obfuscator, classifier, and reconstructor. The obfuscator works as an feature extractor and sensitive information remover to protect users' privacy without decreasing the accuracy of the classifier. The reconstructor is an attacker, and has an opposite objective to reveal the sensitive information. Based on this antagonism, we designed an adversarial training methodology. Experiments showed our framework is qualified to protect users' privacy and achieve a relatively high accuracy on the image classification task.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308–318. ACM, 2016.

Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016.

Reda Bellafqira, Gouenou Coatrieux, Emmanuelle Genin, and Michel Cozic. Secure multilayer perceptron based on homomorphic encryption. *arXiv preprint arXiv:1806.02709*, 2018.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pp. 177–186. Springer, 2010.

Hervé Chabanne, Amaury de Wargny, Jonathan Milgram, Constance Morel, and Emmanuel Prouff. Privacy-preserving classification on deep neural network. *IACR Cryptology ePrint Archive*, 2017: 35, 2017.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pp. 2172–2180, 2016.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Cynthia Dwork. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*, pp. 1–19. Springer, 2008.

Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 776–780. IEEE, 2017.

Ran Gilad-Bachrach, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. Cryptonets: Applying neural networks to encrypted data with high throughput and accuracy. In *International Conference on Machine Learning*, pp. 201–210, 2016.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

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

Phillip Isola, Jun Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. pp. 5967–5976, 2016.

Jianbo Jiao, Wei chi Tu, Shengfeng He, and Rynson W. H. Lau. Formresnet: Formatted residual learning for image restoration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshop (NTIRE)*, 2017.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Meng Li, Liangzhen Lai, Naveen Suda, Vikas Chandra, and David Z Pan. Privynet: A flexible framework for privacy-preserving deep neural network training with a fine-grained privacy control. *arXiv preprint arXiv:1709.06161*, 2017.

Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pp. 106–115. IEEE, 2007.

Wenchao Liu, Feng Pan, Xu An Wang, Yunfei Cao, and Dianhua Tang. Privacy-preserving all convolutional net based on homomorphic encryption. In *International Conference on Network-Based Information Systems*, pp. 752–762. Springer, 2018.

Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. In *null*, pp. 24. IEEE, 2006.

Seyed Ali Osia, Ali Shahin Shamsabadi, Ali Taheri, Kleomenis Katevas, Hamid R Rabiee, Nicholas D Lane, and Hamed Haddadi. Privacy-preserving deep inference for rich user data on the cloud. *arXiv preprint arXiv:1710.01727*, 2017.

Seyed Ali Osia, Ali Taheri, Ali Shahin Shamsabadi, Kleomenis Katevas, Hamed Haddadi, and Hamid R Rabiee. Deep private-feature extraction. *arXiv preprint arXiv:1802.03151*, 2018.

General Data Protection Regulation. Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46. *Official Journal of the European Union (OJ)*, 59(1-88):294, 2016.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Bita Darvish Rouhani, M Sadegh Riazi, and Farinaz Koushanfar. Deepsecure: Scalable provably-secure deep learning. *arXiv preprint arXiv:1705.08963*, 2017.

Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1310–1321. ACM, 2015.

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

Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.