# TIPRDC: Task-Independent Privacy-Respecting Data Crowdsourcing Framework for Deep Learning with Anonymized Intermediate Representations

Ang Li[1], Yixiao Duan[2], Huanrui Yang[1], Yiran Chen[1], Jianlei Yang[2]
[1]Department of Electrical and Computer Engineering, Duke University
[2]School of Computer Science and Engineering, Beihang University
[1]{ang.li630, huanrui.yang, yiran.chen}@duke.edu, [2]{jamesdyx, jianlei}@buaa.edu.cn

## ABSTRACT

The success of deep learning partially benefits from the availability of various large-scale datasets. These datasets are often crowdsourced from individual users and contain private information like gender, age, etc. The emerging privacy concerns from users on data sharing hinder the generation or use of crowdsourcing datasets and lead to hunger of training data for new deep learning applications. One naïve solution is to pre-process the raw data to extract features at the user-side, and then only the extracted features will be sent to the data collector. Unfortunately, attackers can still exploit these extracted features to train an adversary classifier to infer private attributes. Some prior arts leveraged game theory to protect private attributes. However, these defenses are designed for known primary learning tasks, the extracted features work poorly for unknown learning tasks. To tackle the case where the learning task may be unknown or changing, we present *TIPRDC*, a task-independent privacy-respecting data crowdsourcing framework with anonymized intermediate representation. The goal of this framework is to learn a feature extractor that can hide the privacy information from the intermediate representations; while maximally retaining the original information embedded in the raw data for the data collector to accomplish unknown learning tasks. We design a hybrid training method to learn the anonymized intermediate representation: (1) an adversarial training process for *hiding private information from features*; (2) *maximally retain original information* using a neural-network-based mutual information estimator. We extensively evaluate TIPRDC and compare it with existing methods using two image datasets and one text dataset. Our results show that TIPRDC substantially outperforms other existing methods. Our work is the first task-independent privacy-respecting data crowdsourcing framework.

## CCS CONCEPTS

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


*KDD '20, August 23–27, 2020, Virtual Event, USA*

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403125

## KEYWORDS

privacy-respecting data crowdsourcing; anonymized intermediate representations; deep learning

**ACM Reference Format:**
Ang Li[1], Yixiao Duan[2], Huanrui Yang[1], Yiran Chen[1], Jianlei Yang[2]. 2020. TIPRDC: Task-Independent Privacy-Respecting Data Crowdsourcing Framework for Deep Learning with Anonymized Intermediate Representations. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403125

## 1 INTRODUCTION

Deep learning has demonstrated an impressive performance in many applications, such as computer vision [13, 17] and natural language processing [2, 28, 40]. Such success of deep learning partially benefits from various large-scale datasets (e.g., ImageNet [5], MS-COCO [21], etc.), which can be used to train powerful deep neural networks (DNN). The datasets are often crowdsourced from individual users to train DNN models. For example, companies or research institutes that want to implement face recognition systems may collect the facial images from employees or volunteers. However, those data that are crowdsourced from individual users for deep learning applications often contain private information such as gender, age, etc. Unfortunately, the data crowdsourcing process can be exposed to serious privacy risks as the data may be misused by the data collector or acquired by the adversary. It is recently reported that many large companies face data security and user privacy challenges. The data breach of Facebook, for example, raises users' severe concerns on sharing their personal data. These emerging privacy concerns hinder generation or use of large-scale crowdsourcing datasets and lead to hunger of training data of many new deep learning applications. A number of countries are also establishing laws to protect data security and privacy. As a famous example, the new European Union's General Data Protection Regulation (GDPR) requires companies to not store personal data for a long time, and allows users to delete or withdraw their personal data within 30 days. It is critical to design a data crowdsourcing framework to protect the privacy of the shared data while maintaining the utility for training DNN models.

Existing solutions to protect privacy are struggling to balance the tradeoff between privacy and utility. An obvious and widely adopted solution is to transform the raw data into task-oriented features, and users only upload the extracted features to corresponding service providers, such as Google Now [12] and Google Cloud [11]. Even though transmitting only features are generally more secure

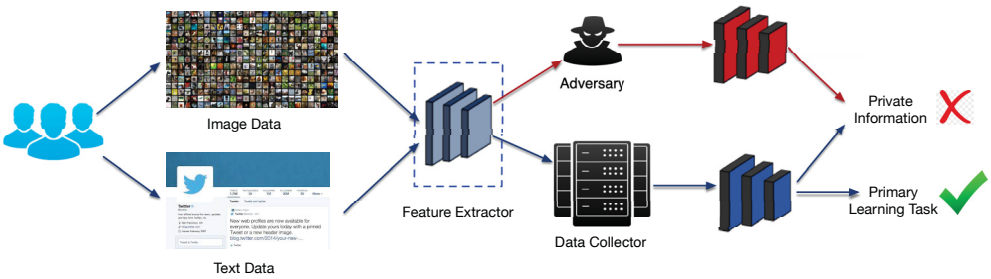



**Figure 1: The overview of TIPRDC.**

than uploading raw data, recent developments in model inversion attacks [6, 7, 24] have demonstrated that adversaries can exploit the acquired features to reconstruct the raw image, and hence the person on the raw image can be re-identified from the reconstructed image. In addition, the extracted features can also be exploited by an adversary to infer private attributes, such as gender, age, etc. Ossia *et al.* [29] move forward by applying dimentionality reduction and noise injection to the features before uploading them to the service provider. However, such approach leads to unignorable utility loss. Inspired by Generative Adversarial Networks (GAN), several adversarial learning approaches [15, 19, 22, 27] have been proposed to learn obfuscated features from raw images. Unfortunately, those solutions are designed for known primary learning tasks, which limits their applicability in the data crowdsourcing where the primary learning task may be unknown or changed when training a DNN model. The need of collecting large-scale crowdsourcing dataset under strict requirement of data privacy and limited applicability of existing solutions motivates us to design a privacy-respecting data crowdsourcing framework: the raw data from the users are locally transformed into an intermediate representation that can remove the private information while retaining the discriminative features for primary learning tasks.

In this work, we propose TIPRDC – a task-independent privacy-respecting data crowdsourcing framework with anonymized intermediate representation. The ultimate goal of this framework is to learn a feature extractor that can remove the privacy information from the extracted intermediate features while maximally retaining the original information embedded in the raw data for primary learning tasks. As Figure 1 shows, participants can locally run the feature extractor and submit only those intermediate representations to the data collector instead of submitting the raw data. The data collector then trains DNN models using these collected intermediate representations, but both the data collector and the adversary cannot accurately infer any protected private information. Compared with existing adversarial learning methods [15, 19, 22, 27], TIPRDC does not require the knowledge of the primary learning task and hence, directly applying existing adversarial training methods becomes impractical. It is challenging to remove all concerned private information that needs to be protected while retaining everything else for unknown primary learning tasks. To address this issue, we design a hybrid learning method to learn the anonymized intermediate representation. The learning purpose is two-folded: (1) hiding private information from features; (2) maximally retaining original information. Specifically, we hide private information from features by performing our proposed privacy adversarial training (PAT) algorithm, which simulates the game between an adversary who makes efforts to infer private attributes from the extracted features and a defender who aims to protect user privacy. The original information are retained by applying our proposed MaxMI algorithm, which aims to maximize the mutual information between the feature of the raw data and the union of the private information and the retained feature.

In summary, our key contributions are the follows:

- To the best of our knowledge, TIPRDC is the first privacy-respecting data crowdsourcing framework for deep learning without the knowledge of any specific primary learning task. By applying TIPRDC, the learned feature extractor can hide private information from features while maximally retaining the information of the raw data.
- We propose a privacy adversarial training algorithm to enable the feature extractor to hide privacy information from features. In addition, we also design the MaxMI algorithm to maximize the mutual information between the raw data and the union of the private information and the retained feature, so that the original information from the raw data can be maximally retained in the feature.
- We quantitatively evaluate the utility-privacy tradeoff with applying TIPRDC on three real-world datasets, including both image and text data. We also compare the performance of TIPRDC with existing solutions.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 presents the problem formulation. Section 4 describes the framework overview and details of core modules. Section 5 evaluates the framework. Section 6 concludes this paper.

## 2 RELATED WORK

**Data Privacy Protection**: Many techniques have been proposed to protect data privacy, most of which are based on various anonymization methods including $k$-anonymity [36], $l$-diversity [24] and $t$-closeness [20]. However, these approaches are designed for protecting sensitive attributes in a static database and hence, are not suitable to our addressed problem – data privacy protection in the online data crowdsourcing for training DNN models. Differential privacy [1, 3, 8, 9, 33, 34, 38] is another widely applied technique to protect privacy of an individual's data record, which provides a strong privacy guarantee. However, the privacy guarantee provided by differential privacy is different from the privacy protection offered by TIPRDC in data crowdsourcing. The goal of differential privacy is to add random noise to a user's true data record

such that two arbitrary true data records have close probabilities to generate the same noisy data record. Compared with differential privacy, our goal is to hide private information from the features such that an adversary cannot accurately infer the protected private information through training DNN models. Osia *et al.* [29] leverage a combination of dimensionality reduction, noise addition, and Siamese fine-tuning to protect sensitive information from features, but it does not offer the tradeoff between privacy and utility in a systematic way.

**Visual Privacy Protection:** Some works have been done to specifically preserve privacy in images and videos. De-identification is a typical privacy-preserving visual recognition approach to alter the raw image such that the identity cannot be visually recognized. There are various techniques to achieve de-identification, such as Gaussian blur [26], identity obfuscation [26], mean shift filtering [39] and adversarial image perturbation [27]. Although those approaches are effective in protecting visual privacy, they all limit the utility of the data for training DNN models. In addition, encryption-based approaches [10, 42] have been proposed to guarantee the privacy of the data, but they require specialized DNN models to directly train on the encrypted data. Unfortunately, such encryption-based solutions prevent general dataset release and introduce substantial computational overhead. All the above practices only consider protecting privacy in specific data format, i.e., image and video, which limit their applicability across diverse data modalities in the real world.

**Tradeoff between Privacy and Utility using Adversarial Networks**: With recent advances in deep learning, several approaches have been proposed to protect data privacy using adversarial networks and simulate the game between the attacker and the defender who defend each other with conflicting utility-privacy goals. Pittaluga *et al.* [32] design an adversarial learning method for learning an encoding function to defend against performing inference for specific attributes from the encoded features. Seong *et al.* [27] introduce an adversarial network to obfuscate the raw image so that the attacker cannot successfully perform image recognition. Wu *et al.* [41] design an adversarial framework to explicitly learn a degradation transform for the original video inputs, aiming to balance between target task performance and the associated privacy budgets on the degraded video. Li *et al.* [19] and Liu *et al.* [22] propose approaches to learn obfuscated features using adversarial networks, and only obfuscated features will be submitted to the service provider for performing inference. Attacker cannot train an adversary classifier using collected obfuscated features to accurately infer a user's private attributes or reconstruct the raw data. The same idea behind the above solutions is that using adversarial networks to obfuscate the raw data or features, in order to defending against privacy leakage. However, those solutions are designed to protect privacy information while targeting some specified learning tasks, such as face recognition, activity recognition, etc. Our proposed TIPRDC provides a more general privacy protection, which does not require the knowledge of the primary learning task.

**Differences between TIPRDC and existing methods:** Compared with prior arts, our proposed TIPRDC has two distinguished features: (1) a more general approach that be applied on different data formats instead of handling only image data or static database; (2) require no knowledge of primary learning tasks.

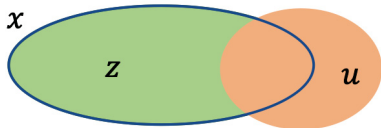



**Figure 2: Optimal outcome of feature extractor $f_\theta(z|x,u)$.**

## 3 PROBLEM FORMULATION

There are three parties involved in the crowdsourcing process: *user*, *adversary*, and *data collector*. Under the strict requirement of data privacy, a data collector offers options to a user to specify any private attribute that needs to be protected. Here we denote the private attribute specified by a user as $u$. According to the requirement of protecting $u$, the data collector will learn a feature extractor $f_\theta(z|x,u)$ that is parameterized by weight $\theta$, which is the core of TIPRDC. The data collector distributes the data collecting request associated with the feature extractor to users. Given the raw data $x$ provided by a user, the feature extractor can locally extract feature $z$ from $x$ while hiding private attribute $u$. Then, only extracted feature $z$ will be shared with the data collector, which can training DNN models for primary learning tasks using collected $z$. An adversary, who may be an authorized internal staff of the data collector or an external hacker, has access to the extracted feature $z$ and aims to infer private attribute $u$ based on $z$. We assume an adversary can train a DNN model via collecting $z$, and then the trained model takes a user's extracted feature $z$ as input and infers the user's private attribute $u$.

The critical challenge of TIPRDC is to learn the feature extractor, which can hide private attribute from features while maximally retaining original information from the raw data. Note that it is very likely that the data and private attribute are somehow correlated. Therefore, we cannot guarantee no information about $u$ will be contained in $z$ unless we enforce it in the objective function when training the feature extractor.

The ultimate goal of the feature extractor $f$ is two-folded:

- **Goal 1:** make sure the extracted features conveys no private attribute;
- **Goal 2:** retain as much information of the raw data as possible to maintain the utility for primary learning tasks.

Formally, Goal 1 can be formulated as:

$$\min_{\theta} I(z;u), \tag{1}$$

where $I(z;u)$ represents the mutual information between $z$ and $u$. On the other hand, Goal 2 can be formulated as:

$$\max_{\theta} I(x;z|u). \tag{2}$$

In order to avoid any potential conflict with the objective of Goal 1, we need to mitigate counting in the information where $u$ and $x$ are correlated. Therefore, the mutual information in Equation 2 is evaluated under the condition of private attribute $u$. Note that

$$I(x;z|u) = I(x;z,u) - I(x;u). \tag{3}$$

Since both $x$ and $u$ are considered fixed under our setting, $I(x;u)$ will stay as a constant during the optimization process of feature extractor $f_\theta$. Therefore, we can safely rewrite the objective of Goal 2 as:

$$\max_{\theta} I(x;z,u), \tag{4}$$

which is to maximize the mutual information between $x$ and joint distribution of $z$ and $u$.

As Figure 2 illustrates, we provide an intuitive demonstration of the optimal outcome of the feature extractor using a Venn diagram. Goal 1 is fulfilled as no overlap exist between $z$ (green area) and $u$ (orange area); and Goal 2 is achieved as $z$ and $u$ jointly (all colored regions) cover all the information of $x$ (blue circle).

It is widely accepted in the previous works that precisely calculating the mutual information between two arbitrary distributions are likely to be infeasible [31]. As a result, we replace the mutual information objectives in Equation 1 and 4 with their upper and lower bounds for effective optimization. For Goal 1, we utilize the mutual information upper bound derived in [35] as:

$$I(z;u) \leq \mathbb{E}_{q_\theta(z)} D_{KL}(q_\theta(u|z)||p(u)), \tag{5}$$

for any distribution $p(u)$. Note that the term $q_\theta(u|z)$ in Equation (5) is hard to estimate and hence we instead replace the KL divergence term with its lower bound by introducing a conditional distribution $p_\Psi(u|z)$ parameterized with $\Psi$. It was shown in [35] that:

$$\begin{aligned} &\mathbb{E}_{q_\theta(z)}\left[\log p_\Psi(u|z) - \log p(u)\right] \\ &\leq \mathbb{E}_{q_\theta(z)} D_{KL}(q_\theta(u|z)||p(u)) \end{aligned} \tag{6}$$

Hence, the Equation 1 can be rewritten as an adversarial training objective function:

$$\min_{\theta}\max_{\Psi} \mathbb{E}_{q_\theta(z)}\left[\log p_\Psi(u|z) - \log p(u)\right], \tag{7}$$

As $\log p(u)$ is a constant number that is independent of $z$, Equation 7 can be further simplified to:

$$\max_{\theta}\min_{\Psi} -\mathbb{E}_{q_\theta(z)}\left[\log p_\Psi(u|z)\right], \tag{8}$$

which is the cross entropy loss of predicting $u$ with $p_\Psi(u|z)$, i.e., $CE\left[p_\Psi(u|z)\right]$. This objective function can be interpreted as an adversarial game between an adversary $p_\Psi$ who tries to infer $u$ from $z$ and a defender $q_\theta$ who aims to protect the user privacy.

For Goal 2, we adopt the previously proposed Jensen-Shannon mutual information estimator [14, 25] to estimate the lower bound of the mutual information I(x; z,u). The lower bound is formulated as follows:

$$\begin{aligned} &\mathcal{I}(x;z,u) \geq \mathcal{I}^{(JSD)}_{\theta,\omega}(x;z,u) \\ &:= \mathbb{E}_x\left[-sp(-E_\omega(x;f_\theta(x),u))\right] - \mathbb{E}_{x,x'}\left[sp(E_\omega(x';f_\theta(x),u))\right], \end{aligned} \tag{9}$$

where $x'$ is an random input data sampled independently from the same distribution of $x$, $sp(z) = log(1+e^z)$ is the softplus function and $E_\omega$ is a discriminator modeled by a neural network with parameters $\omega$. Hence, to maximally retain the original information, the feature extractor and the mutual information estimator can be optimized using Equation 10:

$$\max_{\theta}\max_{\omega} \mathcal{I}^{(JSD)}_{\theta,\omega}(x;z,u). \tag{10}$$

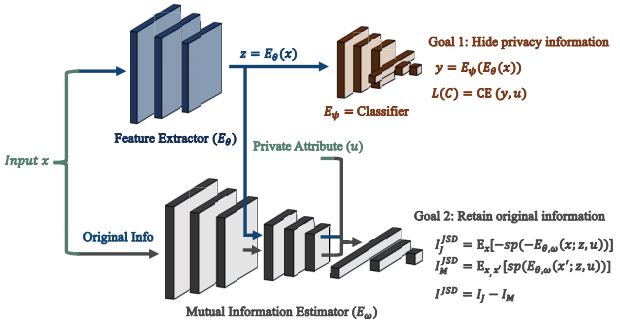



**Figure 3: The hybrid learning method for training the feature extractor.**

Finally, combining Equation 8 and 10, the objective function of training the feature extractor can be formulated as:

$$\max_{\theta}(\lambda \min_{\Psi} CE\left[p_\Psi(u|z)\right] + (1-\lambda)\max_{\omega} \mathcal{I}^{(JSD)}_{\theta,\omega}(x;z,u)), \tag{11}$$

where $\lambda \in [0,1]$ serves as a utility-privacy budget. A larger $\lambda$ indicates a stronger privacy protection, while a smaller $\lambda$ allowing more original information to be retained in the extracted features.

# 4 DESIGN OF TIPRDC

## 4.1 Overview

The critical module of TIPRDC is the feature extractor. As presented in Section 3, there are two goals for learning the feature extractor, such that it can hide private attribute from features while retaining as much information of the raw data as possible to maintain the utility for primary learning task. To this end, we design a hybrid learning method to train the feature extractor, including the privacy adversarial training (PAT) algorithm and the MaxMI algorithm. In particular, we design the PAT algorithm, which simulates the game between an adversary who makes efforts to infer private attributes from the extracted features and a defender who aims to protect user privacy. By applying PAT to optimize the feature extractor, we enforce the feature extractor to hide private attribute $u$ from extracted features $z$, which is goal 1 introduced in Section 3. Additionally, we propose the MaxMI algorithm to achieve goal 2 presented in Section 3. By performing MaxMI to train the feature extractor, we can enable the feature extractor to maximize the mutual information between the information of the raw data $x$ and the joint information of the private attribute $u$ and the extracted feature $z$.

As Figure 3 shows, there are three neural network modules in the hybrid learning method: *feature extractor*, *adversarial classifier* and *mutual information estimator*. The feature extractor is the one we aim to learn by performing the proposed hybrid learning algorithm. The adversarial classifier simulates an adversary in the PAT algorithm, aiming to infer private attribute $u$ from the eavesdropped features. The mutual information estimator is adopted in MaxMI algorithm to measure the mutual information between the raw data $x$ and the joint distribution of the private attribute $u$ and the extracted feature $z$. All three modules are end-to-end trained using our proposed hybrid learning method.

Before presenting the details of each algorithm, we give the following notations. As same as presented in Section 3, we adopt $x$, $u$ and $z$ as the raw data, the private attribute and the extracted feature, respectively. We denote $E_\theta$ as the feature extractor that is parameterized with $\theta$, and then $z$ can be expressed as $E_\theta(x)$. Let $E_\Psi$ represent the classifier, where $\Psi$ indicates the parameter set of the classifier. We adopt $y = E_\Psi(E_\theta(x))$ to denote the prediction generated by $E_\Psi$. Let $E_\omega$ denotes the mutual information estimator, where $\omega$ represents the parameter set of the mutual information estimator.

## 4.2 Privacy Adversarial Training Algorithm

We design the PAT algorithm to achieve goal 1, enabling the feature extractor to hide $u$ from $E_\theta(x)$. The PAT algorithm is designed by simulating the game between an adversary who makes efforts to infer private attributes from the extracted features and a defender who aims to protect user privacy. We can apply any architecture to both the feature extractor and the classifier based on the requirement of data format and the primary learning task. The performance of the classifier ($C$) is measured using the cross-entropy loss function as:

$$\mathcal{L}(C) = CE(y, u) = CE(E_\Psi(E_\theta(x)), u), \tag{12}$$

where $CE[\cdot]$ stands for the cross entropy loss function. When we simulate an adversary who tries to enhance the accuracy of the adversary classifier as high as possible, the classifier needs to be optimized by minimizing the above loss function as:

$$\Psi = \arg\min_{\Psi} \mathcal{L}(C). \tag{13}$$

On the contrary, when defending against private attribute leakage, we train the feature extractor in PAT that aims to degrade the performance of the classifier. Consequently, the feature extractor can be trained using Equation 14 when simulating a defender:

$$\theta = \arg\max_{\theta} \mathcal{L}(C). \tag{14}$$

Based on Equation 13 and 14, the feature extractor and the classifier can be jointly optimized using Equation 15:

$$\theta, \Psi = \arg\max_{\theta} \min_{\Psi} \mathcal{L}(C), \tag{15}$$

which is consistent with Equation 8.

## 4.3 MaxMI Algorithm

For goal 2, we propose MaxMI algorithm to make the feature extractor retain as much as information from the raw data as possible, in order to maintain the high utility of the extracted features. Specifically, the Jensen-Shannon mutual information estimator [14, 25] is adopted to measure the lower bound of the mutual information $I(x; z, u)$. Here we adopt $E_\omega$ as the the Jensen-Shannon mutual information estimator, and we can rewrite Equation 9 as:

$$\begin{aligned} &\mathcal{I}(x; z, u) \geq \mathcal{I}^{(JSD)}_{\theta,\omega}(x; z, u) \\ &:= \mathbb{E}_x\left[-sp(-E_\omega(x, E_\theta(x), u))\right] - \mathbb{E}_{x,x'}\left[sp(E_\omega(x', E_\theta(x), u))\right], \end{aligned} \tag{16}$$

where $x'$ is an random input data sampled independently from the same distribution of $x$, $sp(z) = log(1 + e^z)$ is the softplus function. Hence, to maximally retain the original information, the feature extractor and the mutual information estimator can be optimized using Equation 17:

$$\theta, \omega = \arg\max_{\theta} \max_{\omega} \mathcal{I}^{(JSD)}_{\theta,\omega}(x; z, u), \tag{17}$$

which is consistent with Equation 10. Considering the difference of $x$, $z$ and $u$ in dimensionality, we feed them into the $E_\omega$ from different layers as illustrated in Figure 3. For example, the private attribute $u$ may be a binary label, which is represented by one bit. However, $x$ may be high dimensional data (e.g., image), and hence it is not reasonable feed both $x$ and $u$ from the first layer of $E_\omega$.

## 4.4 Hybrid Learning Method

Finally, the feature extractor is trained by alternatively performing PAT algorithm and MaxMI algorithm. As aforementioned, we also introduce an utility-privacy budget $\lambda$ to balance the tradeoff between protecting privacy and retaining the original information. Therefore, combining Equation 15 and 17, the objective function of training the feature extractor can be formulated as:

$$\theta, \Psi, \omega = \arg\max_{\theta}(\lambda \min_{\Psi} \mathcal{L}(C) + (1-\lambda) \max_{\omega} \mathcal{I}^{(JSD)}_{\theta,\omega}(x; z, u)). \tag{18}$$

Algorithm 1 summarizes the hybrid learning method of TIPRDC. Before performing the hybrid learning, we first jointly pretrain the feature extractor and the adversarial classifier normally without adversarial objective to obtain the best performance on classifying a specific private attribute. Then within each training batch, we first perform PAT algorithm and MaxMI algorithm to update $\Psi$ and $\omega$, respectively. Then, the feature extractor will be updated according to Equation 18.

**Algorithm 1** Hybrid Learning Method

**Input:** Dataset $\mathcal{D}$
**Output:** $\theta$

```
for every epoch do
    for every batch do
        L(C) → update ψ (performing PAT)
        −I^JSD_{θ,ω}(x; z, u) → update ω (performing MaxMI)
        −λL(C) − (1 − λ)I^JSD_{θ,ω}(x; z, u) → update θ
    end for
end for
```

# 5 EVALUATION

In this section, we evaluate TIPRDC's performance on three real-world datasets, with a focus on the utility-privacy tradeoff. We also compare TIPRDC with existing solutions proposed in the literature and visualize the results.

## 5.1 Experiment Setup

We evaluate TIPRDC, especially the learned feature extractor, on two image datasets and one text dataset. We implement TIPRDC with PyTorch, and train it on a server with 4×NVIDIA TITAN RTX GPUs. We apply mini-batch technique in training with a batch size of 64, and adopt the AdamOptimizer [16] with an adaptive learning rate in the hybrid learning procedure. The architecture configurations of each module are presented in Table 1 and 2. For evaluating

the performance, given a primary learning task, a simulated data collector trains a normal classifier using features processed by the learned feature extractor, and such normal classifier has the same architecture of the classifier presented in Table 1 and 2. The utility and privacy of the extracted features $E_\theta(x)$ are evaluated by the classification accuracy of primary learning tasks and specified private attribute, respectively.

We adopt CelebA [23], LFW [18] and the dialectal tweets dataset (DIAL) [4] for the training and testing of TIPRDC. CelebA consists of more than 200K face images. Each face image is labeled with 40 binary facial attributes. The dataset is split into 160K images for training and 40K images for testing. LFW consists of more than 13K face images, and each face image is labeled with 16 binary facial attributes. We split LFW into 10K images for training and 3K images for testing. DIAL consists of 59.2 million tweets collected from 2.8 million users, and each tweet is annotated with three binary attributes. DIAL is split into 48 million tweets for training and 11.2 million tweets for testing.

**Table 1: The architecture configurations of each module for CelebA and LFW.**

| Feature Extractor | Classifier | MI Estimator |
|---|---|---|
| 2×conv3-64<br>maxpool | 3×conv3-256<br>maxpool | 3×conv3-64<br>maxpool |
| 2×conv3-128<br>maxpool | 3×conv3-512<br>maxpool | 2×conv3-128<br>maxpool |
| | 3×conv3-512<br>maxpool | 3×conv3-256<br>maxpool |
| | 2×fc-4096<br>fc-label length | 3×conv3-512<br>maxpool |
| | | 3×conv3-512<br>maxpool |
| | | fc-4096<br>fc-512<br>fc-1 |

**Table 2: The architecture configurations of each module for DIAL.**

| Feature Extractor | Classifier | MI Estimator |
|---|---|---|
| embedding-300 | 2×lstm-300 | embedding-300 |
| lstm-300 | fc-150 | lstm-300 |
| | fc-label length | 2×lstm-300 |
| | | fc-150<br>fc-1 |

## 5.2 Comparison Baselines

We select four types of data privacy-preserving baselines [22], which have been widely applied in the literature, and compare them with TIPRDC. The details settings of the baseline solutions are presented as below.

- **Noisy** method perturbs the raw data $x$ by adding Gaussian noise $\mathcal{N}(0, \sigma^2)$, where $\sigma$ is set to 40 according to [22]. The noisy data $\bar{x}$ will be delivered to the data collector. The Gaussian noise injected to the raw data can provide strong guarantees of differential privacy using less local noise. This scheme has been widely applied in federated learning [30, 37].
- **DP** approach injects Laplace noise the raw data $x$ with diverse privacy budgets {0.1, 0.2, 0.5, 0.9}, which is a typical differential privacy method. The noisy data $\bar{x}$ will be submitted to the data collector.
- **Encoder** learns the latent representation of the raw data $x$ using a DNN-based encoder. The extracted features $z$ will be uploaded to the data collector.
- **Hybrid** method [29] further perturbs the above encoded features by performing principle components analysis (PCA) and adding Laplace noise with varying noise factors privacy budgets {0.1, 0.2, 0.5, 0.9}.

## 5.3 Evaluations on CelebA and LFW

**Comparison of utility-privacy tradeoff:** We compare the utility-privacy tradeoff offered by TIPRDC with four privacy-preserving baselines. In our experiments, we set 'young' and 'gender' as the private labels to protect in CelebA, and consider detecting 'gray hair' and 'smiling' as the primary learning tasks to evaluate the utility. With regard to LFW, we set 'gender' and 'Asian' as the private labels, and choose recognizing 'black hair' and 'eyeglass' as the classification tasks. Figure 4 summarizes the utility-privacy tradeoff offered by four baselines and TIPRDC. Here we evaluate TIPRDC with four discrete choices of $\lambda \in \{1, 0.9, 0.5, 0\}$.

As Figure 4 shows, although TIPRDC cannot always outperform the baselines in both utility and privacy, it still achieve the best utility-privacy tradeoff under most experiment settings. For example, in Figure 4 (h), TIPRDC achieves the best tradeoff by setting $\lambda = 0.9$. Specifically, the classification accuracy of 'Asian' on LFW is 55.31%, and the accuracy of 'eyeglass' is 86.88%. This demonstrates that TIPRDC can efficiently protect privacy while maintaining high utility of extracted features.

In other four baselines, Encoder method can maintain a good utility of the extracted features, but it fails to protect privacy due to the high accuracy of private labels achieved by the adversary. Noisy, DP and Hybrid methods offer strong privacy protection with sacrificing the utility.

**Impact of the utility-privacy budget $\lambda$:** An important step in the hybrid learning procedure (see Equation 18) is to determine the utility-privacy budget $\lambda$. To determine the optimal $\lambda$, we evaluate the utility-privacy tradeoff on CelebA and LFW by setting different $\lambda$. Specifically, we evaluate the impact of $\lambda$ with four discrete choices of $\lambda \in \{1, 0.9, 0.5, 0\}$. The private labels and primary learning tasks in CelebA and LFW are set as same as the above experiments.

As Figure 6 illustrates, the classification accuracy of primary learning tasks will increase with a smaller $\lambda$, but the privacy protection will be weakened. Such phenomenon is reasonable, since the smaller $\lambda$ means hiding less privacy information in features but retaining more original information from the raw data according to Equation 18. For example, in Figure 6 (a), the classification accuracy of 'gray hair' on CelebA is 84.36% with $\lambda = 1$ and increases to 91.52% by setting $\lambda = 0$; the classification accuracy of 'young' is 65.63% and 81.85% with decreasing $\lambda$ from 1 to 0, respectively. Overall, $\lambda = 0.9$ is an optimal utility-privacy budget for experiment settings in both CelebA and LFW.

We further visualize how different options of $\lambda$ influence the utility maintained by the learned feature extractor. This is done

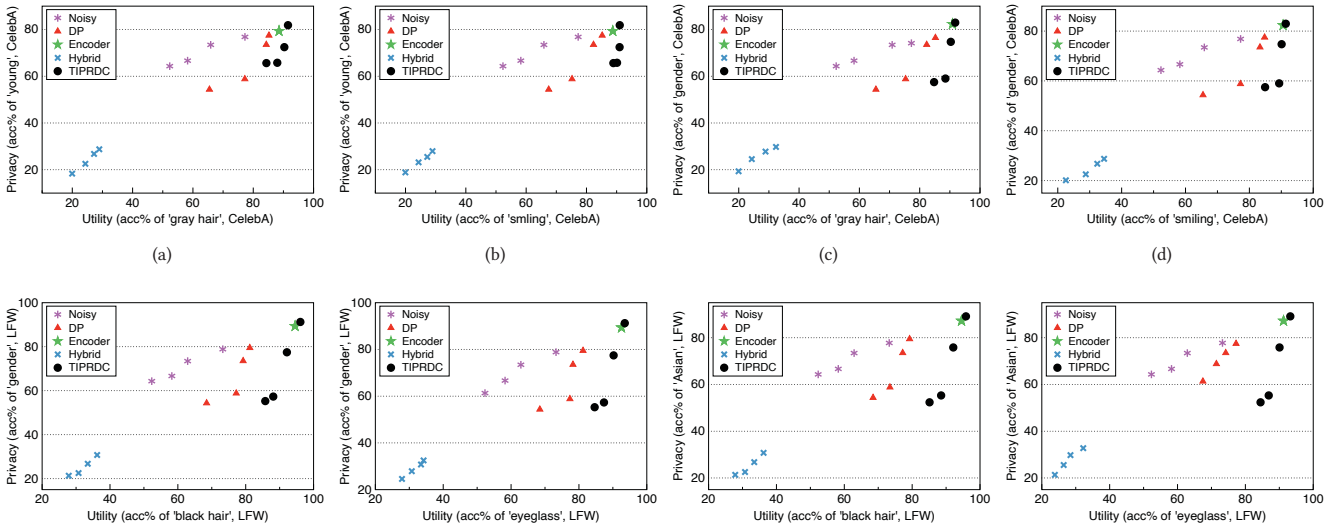
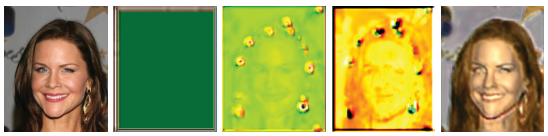



**Figure 4: Utility-privacy tradeoff comparison of TIPRDC with four baselines on CelebA and LFW.**



(a) raw image (b) $\lambda = 1$ (c) $\lambda = 0.9$ (d) $\lambda = 0.5$ (e) $\lambda = 0$

**Figure 5: Visualize the impact of the utility-privacy budget $\lambda$ when protecting 'gender' in CelebA.**

by training a decoder with the reversed architecture of the feature extractor, and then the decoder aims to reconstruct the raw data by taking the extracted feature as input. Here we adopt the setting in Figure 6(b) as an example, where 'gender' is protected when training the feature extractor. As Figure 5 shows, decreasing $\lambda$ allows a more informative image to be reconstructed. This means more information is retained in the extracted feature with a smaller $\lambda$, which is consist with the results shown in Figure 6(c).

Additionally, if we compare Figure 6(c) vs. Figure 6(a-b) and Figure 6(f) vs. Figure 6(d-e), it can be observed that protecting more private attributes leads to slight degradation in utility with slightly enhanced privacy protection under a particular $\lambda$. For example, given $\lambda = 0.9$, the accuracy of 'smiling' slightly decreases from 90.13% in Figure 6(a) and 89.33% in Figure 6(c) to 88.16%. The accuracy of 'young' slightly decreases from 65.77% in Figure 6(a) to 64.85% in Figure 6(c). The reason is that the feature related to the private attributes has some intrinsic correlations to the feature related to the primary learning tasks. Therefore, more correlated features may be hidden if more private attributes need to be protected.

**Effectiveness of privacy protection:** We quantitatively evaluate the effectiveness of privacy protection offered by TIPRDC by simulating an adversary to infer the private attribute through training a classifier. As presented in Table 1, we adopt a default architecture for simulating the adversary's classifier. However, an adversary may train the classifier with different architectures. We

**Table 3: The ablation study with different architecture configurations of the adversary classifier.**

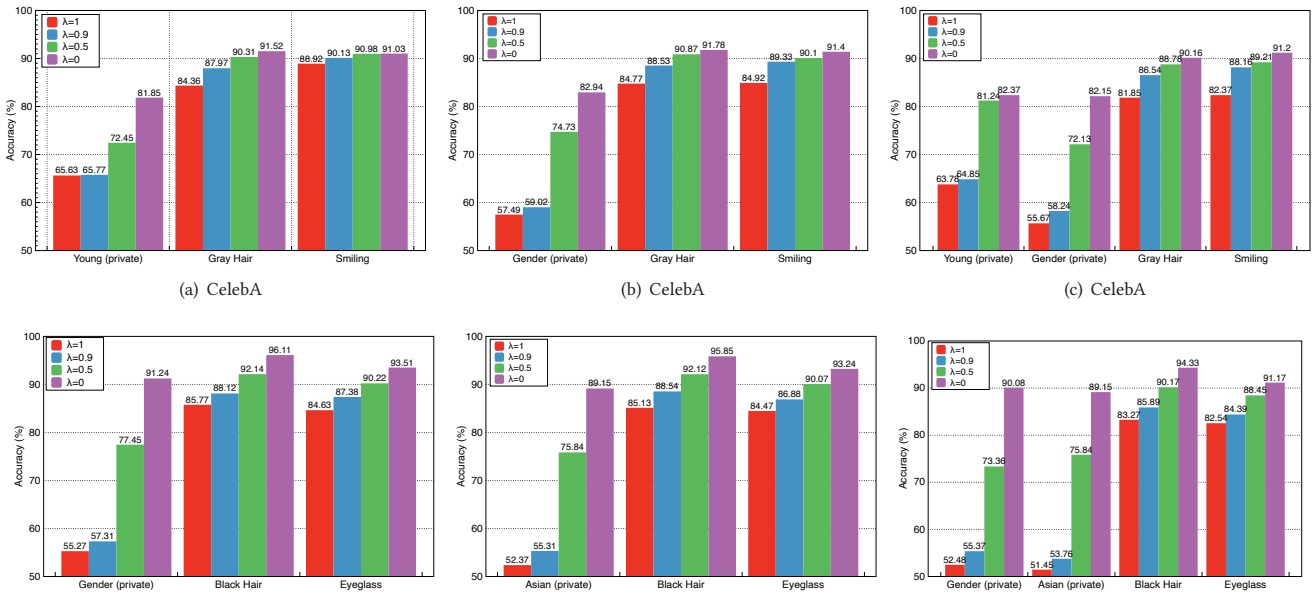
| V-CL#16 | V-CL#19 | Res-CL |
|---|---|---|
| Input Feature Maps ($54 \times 44 \times 128$) | | |
| 3×conv3-256<br>maxpool | 4×conv3-256<br>maxpool | $\begin{bmatrix} 3\times3(2), & 128 \\ 3\times3, & 128 \end{bmatrix}$<br>$\begin{bmatrix} 3\times3, & 128 \\ 3\times3, & 128 \end{bmatrix}$ |
| 3×conv3-512<br>maxpool | 4×conv3-512<br>maxpool | $\begin{bmatrix} 3\times3, & 256 \\ 3\times3, & 256 \end{bmatrix} \times 2$ |
| 3×conv3-512<br>maxpool | 4×conv3-512<br>maxpool | $\begin{bmatrix} 3\times3, & 512 \\ 3\times3, & 512 \end{bmatrix} \times 2$ |
| 2×fc-4096<br>fc-label length | | avgpool<br>fc-label length |

**Table 4: The classification accuracy of 'gender' on CelebA with different adversary classifiers ($\lambda = 0.9$).**

| Training Classifier | Adversary Classifier | | | |
|---|---|---|---|---|
| | Default architecture in Table 1 | V-CL#16 | V-CL#19 | Res-CL |
| Default architecture in Table 1 | 59.02% | 59.83% | 60.77% | 61.05% |

implement three additional classifiers as an adversary in our experiments. The architectural configurations of those classifiers are presented in Table 3. We train those classifiers on CelebA by considering recognizing 'smiling' as the primary learning task, and 'gender' as the private attribute that needs to be protected. Table 4 presents the average classification accuracy for adversary classifiers on testing data. The results show that although we apply a default architecture for simulating the adversary classifier when training the feature extractor, the trained feature extractor can effectively defend against privacy leakage no matter what kinds of architecture are adopted by an adversary in the classifier design.

**Evaluate the transferability of TIPRDC:** The data collector usually trains the feature extractor of TIPRDC before collecting the data from users. Hence, the transferability of the feature extractor




(a) CelebA (b) CelebA (c) CelebA

(d) LFW (e) LFW (f) LFW

**Figure 6: The impact of the utility-privacy budget $\lambda$ on CelebA and LFW.**

**Table 5: Evaluate the transferability of TIPRDC with cross-dataset experiments ($\lambda = 0.9$).**

| Training Dataset | Test Dataset | 'gender' | 'black hair' |
|---|---|---|---|
| LFW | CelebA | 59.73% | 87.15% |
| LFW | LFW | 57.31% | 88.12% |
| CelebA | LFW | 56.87% | 89.27% |
| CelebA | CelebA | 58.82% | 88.98% |

determines the usability of TIPRDC. We evaluate the transferability of TIPRDC by performing cross-dataset evaluations. Specifically, we train the feature extractor of TIPRDC using either CelebA or LFW dataset and test the utility-privacy tradeoff on the other dataset. In this experiment, we choose recognizing 'black hair' as the primary learning task, and 'gender' as the private attribute that needs to be protected. As Table 5 illustrates, the feature extractor that is trained using one dataset can still effectively defend against private attribute leakage on the other dataset, while maintaining the classification accuracy of the primary learning task. For example, if we train the feature extractor using CelebA and then test it on LFW, the accuracy of 'gender' decreases to 56.87% compared with 57.31% by directly training the feature extractor using LFW. The accuracy of 'black hair' marginally increases to 89.27% from 88.12%. The reason is that CelebA offers a larger number of training data so that the feature extractor can be trained for a better performance. Although there is a marginal performance drop, the feature extractor that is trained using LFW still works well on CelebA. The cross-dataset evaluations demonstrate good transferability of TIPRDC.

## 5.4 Evaluation on DIAL

To quantitatively evaluate the utility-privacy tradeoff of TIPRDC on DIAl, we choose 'race' as the private attribute that needs to be protected and predicting 'mentioned' as the primary learning task. The binary mention task is to determine if a tweet mentions another user, i.e., classifying conversational vs. non-conversational tweets. Similar to the experiment settings in CelebA and LFW, we evaluate the utility-privacy tradeoff on DIAL by setting different $\lambda$ with four discrete choices of $\lambda \in \{1, 0.9, 0.5, 0\}$.

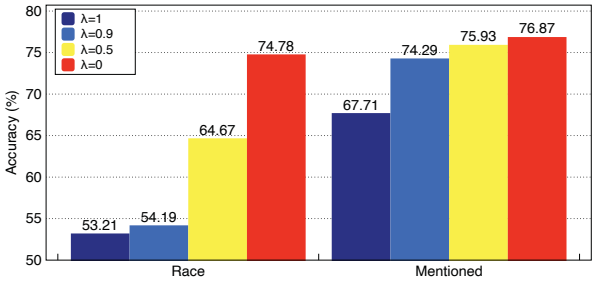



**Figure 7: The impact of the utility-privacy budget $\lambda$ on DIAL.**

As Figure 7 shows, the classification accuracy of primary learning tasks will increase with a smaller $\lambda$, but the privacy protection will be weakened, showing the same phenomenon as the evaluations on CelebA and LFW. For example, the classification accuracy of 'mentioned' is 67.71% with $\lambda = 1$ and increases to 76.87% by setting $\lambda = 0$, and the classification accuracy of 'race' increases by 21.57% after changing $\lambda$ from 1 to 0.

## 6 CONCLUSION

We proposed a task-independent privacy-respecting data crowdsourcing framework TIPRDC. A feature extractor is learned to hide privacy information features and maximally retain original information from the raw data. By applying TIPRDC, a user can locally extract features from the raw data using the learned feature extractor, and the data collector will acquire the extracted features

only to train a DNN model for the primary learning tasks. Evaluations on three benchmark datasets show that TIPRDC attains a better privacy-utility tradeoff than existing solutions. The cross-dataset evaluations on CelebA and LFW shows the transferability of TIPRDC, indicating the practicability of proposed framework.

## ACKNOWLEDGMENTS

This work was supported in part by NSF-1822085 and NSF IUCRC for ASIC membership from Ergomotion. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF and their contractors.

## REFERENCES

[1] Brendan Avent, Aleksandra Korolova, David Zeber, Torgeir Hovden, and Benjamin Livshits. 2017. {BLENDER}: Enabling local search with a hybrid differential privacy model. In *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 747–764.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[3] Raef Bassily and Adam Smith. 2015. Local, private, efficient protocols for succinct histograms. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*. 127–135.

[4] Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. *arXiv preprint arXiv:1608.08868* (2016).

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[6] Alexey Dosovitskiy and Thomas Brox. 2016. Generating images with perceptual similarity metrics based on deep networks. In *Advances in neural information processing systems*. 658–666.

[7] Alexey Dosovitskiy and Thomas Brox. 2016. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4829–4837.

[8] John C Duchi, Michael I Jordan, and Martin J Wainwright. 2013. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. IEEE, 429–438.

[9] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*. 1054–1067.

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

[11] Google. 2018. Data Preparation. https://cloud.google.com/ml-engine/docs/tensorflow/data-prep.

[12] Google. 2018. Google Now Launcher. https://en.wikipedia.org/wiki/Google_Now.

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

[14] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2018. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670* (2018).

[15] Tae-hoon Kim, Dongmin Kang, Kari Pulli, and Jonghyun Choi. 2019. Training with the invisibles: Obfuscating images to share safely for learning visual recognition models. *arXiv preprint arXiv:1901.00098* (2019).

[16] Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv.org* (Dec. 2014), arXiv:1412.6980. arXiv:cs.LG/1412.6980

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[18] Neeraj Kumar, Alexander C Berg, Peter N Belhumeur, and Shree K Nayar. 2009. Attribute and simile classifiers for face verification. In *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 365–372.

[19] Ang Li, Jiayi Guo, Huanrui Yang, and Yiran Chen. 2019. Deepobfuscator: Adversarial training framework for privacy-preserving image classification. *arXiv preprint arXiv:1909.04126* (2019).

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

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[22] Sicong Liu, Junzhao Du, Anshumali Shrivastava, and Lin Zhong. 2019. Privacy Adversarial Network: Representation Learning for Mobile Data Privacy. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 4 (2019), 1–18.

[23] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

[24] Aravindh Mahendran and Andrea Vedaldi. 2015. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5188–5196.

[25] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. 2016. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*. 271–279.

[26] Seong Joon Oh, Rodrigo Benenson, Mario Fritz, and Bernt Schiele. 2016. Faceless person recognition: Privacy implications in social media. In *European Conference on Computer Vision*. Springer, 19–35.

[27] Seong Joon Oh, Mario Fritz, and Bernt Schiele. 2017. Adversarial image perturbation for privacy protection a game theory perspective. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, 1491–1500.

[28] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499* (2016).

[29] Seyed Ali Osia, Ali Shahin Shamsabadi, Sina Sajadmanesh, Ali Taheri, Kleomenis Katevas, Hamid R Rabiee, Nicholas D Lane, and Hamed Haddadi. 2020. A hybrid deep learning architecture for privacy-preserving mobile analytics. *IEEE Internet of Things Journal* (2020).

[30] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. 2018. Scalable private learning with pate. *arXiv preprint arXiv:1802.08908* (2018).

[31] Xue Bin Peng, Angjoo Kanazawa, Sam Toyer, Pieter Abbeel, and Sergey Levine. 2018. Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow. *arXiv preprint arXiv:1810.00821* (2018).

[32] Francesco Pittaluga, Sanjeev Koppal, and Ayan Chakrabarti. 2019. Learning privacy preserving encodings through adversarial training. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 791–799.

[33] Zhan Qin, Yin Yang, Ting Yu, Issa Khalil, Xiaokui Xiao, and Kui Ren. 2016. Heavy hitter estimation over set-valued data with local differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. 192–203.

[34] Adam Smith, Abhradeep Thakurta, and Jalaj Upadhyay. 2017. Is interaction necessary for distributed private learning?. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 58–77.

[35] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. 2018. Learning controllable fair representations. *arXiv preprint arXiv:1812.04218* (2018).

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

[37] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, Rui Zhang, and Yi Zhou. 2019. A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*. 1–11.

[38] Tianhao Wang, Jeremiah Blocki, Ninghui Li, and Somesh Jha. 2017. Locally differentially private protocols for frequency estimation. In *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 729–745.

[39] Thomas Winkler, Adám Erdélyi, and Bernhard Rinner. 2014. TrustEYE. M4: protecting the sensor—not the camera. In *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 159–164.

[40] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016).

[41] Zhenyu Wu, Zhangyang Wang, Zhaowen Wang, and Hailin Jin. 2018. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 606–624.

[42] Ryo Yonetani, Vishnu Naresh Boddeti, Kris M Kitani, and Yoichi Sato. 2017. Privacy-preserving visual learning using doubly permuted homomorphic encryption. In *Proceedings of the IEEE International Conference on Computer Vision*. 2040–2050.