Task-Agnostic Privacy-Preserving Representation Learning for Federated Learning against Attribute Inference Attacks

Caridad Arroyo Arevalo[1], **Sayedeh Leila Noorbakhsh**[1], **Yun Dong**[2], **Yuan Hong**[3], **Binghui Wang**[1]

[1]Illinois Institute of Technology
[2]Benedictine University
[3]University of Connecticut

carroyoarevalo@hawk.iit.edu, snoorbakhsh@hawk.iit.edu, ydong@ben.edu, yuan.hong@uconn.edu, bwang70@iit.edu

Abstract

Federated learning (FL) has been widely studied recently due to its property to collaboratively train data from different devices without sharing the raw data. Nevertheless, recent studies show that an adversary can still be possible to infer private information about devices' data, e.g., sensitive attributes such as income, race, and sexual orientation. To mitigate the attribute inference attacks, various existing privacy-preserving FL methods can be adopted/adapted. However, all these existing methods have key limitations: they need to know the FL task in advance, or have intolerable computational overheads or utility losses, or do not have provable privacy guarantees.

We address these issues and design a task-agnostic privacy-preserving presentation learning method for FL (**TAPPFL**) against attribute inference attacks. TAPPFL is formulated via information theory. Specifically, TAPPFL has two mutual information goals, where one goal learns task-agnostic data representations that contain the least information about the private attribute in each device's data, and the other goal ensures the learnt data representations include as much information as possible about the device data to maintain FL utility. We also derive privacy guarantees of TAPPFL against worst-case attribute inference attacks, as well as the inherent tradeoff between utility preservation and privacy protection. Extensive results on multiple datasets and applications validate the effectiveness of TAPPFL to protect data privacy, maintain the FL utility, and be efficient as well. Experimental results also show that TAPPFL outperforms the existing defenses.

Introduction

The emerging collaborative data analysis using federated learning (FL) (McMahan et al. 2017a) aims to address the data insufficiency problem, and has a great potential to protect data privacy as well. In FL, the participating devices train their data locally, and only share the trained models, instead of the raw data, with a center server (e.g., cloud). The server updates its global model by aggregating the received device models, and broadcasts the updated global model to all participating devices such that all devices *indirectly* use all data from other devices. FL has been deployed by many companies such as Google (Google Federated Learning 2022), Microsoft (Microsoft Federated Learning 2022), IBM (IBM Federated Learning 2022), and Alibaba (Alibaba Federated Learning 2023), and applied in various privacy-sensitive applications, including on-device item ranking (McMahan et al. 2017a), content suggestions for on-device keyboards (Bonawitz et al. 2019), next word prediction (Li et al. 2020b), health monitoring (Rieke et al. 2020), and medical imaging (Kaissis et al. 2020). However, recent works show, though only sharing device models, it is still possible for an adversary (e.g., an honest-but-curious server) to perform the *attribute inference attack* (Jia et al. 2017; Aono et al. 2017; Ganju et al. 2018; Melis et al. 2019; Dang et al. 2021; Wainakh et al. 2022)—i.e., inferring the private/sensitive information (e.g., a person's gender, race, sexual orientation, income) of device's data. Hence, designing privacy-preserving FL mechanisms to defend against the attribute inference attack is important and necessary.

To mitigate the issue, various existing privacy-preserving FL methods can be adopted/adapted, including *multi-party computation (MPC)* (Danner and Jelasity 2015; Mohassel and Zhang 2017; Bonawitz et al. 2017; Melis et al. 2019), *adversarial training (AT)* (Madras et al. 2018; Liu et al. 2019; Li et al. 2020a; Oh, Fritz, and Schiele 2017; Kim et al. 2019), *model compression (MC)* (Zhu, Liu, and Han 2019), and *differential privacy (DP)* (Pathak, Rane, and Raj 2010; Shokri and Shmatikov 2015; Hamm, Cao, and Belkin 2016; McMahan et al. 2018; Geyer, Klein, and Nabi 2017). However, these existing methods have key limitations, thus narrowing their applicability (see Table 1). Specifically, MPC and AT methods know the FL task in advance. However, this cannot be achieved in real-world unsupervised learning applications. MPC methods also have intolerable computational overheads and AT methods do not have provable privacy guarantees. MC and DP methods are task-agnostic, but both of them result in high utility losses (see Figure 3).

In this paper, we aim to design a privacy-preserving FL mechanism against attribute inference attacks (termed **TAPPFL**) that is *task-agnostic*, *efficient*, *accurate*, and have *privacy guarantees* as well. Our main idea is to learn federated privacy-preserving representations based on information theory. Specifically, we formulate TAPPFL via two mutual information (MI) goals, where one MI goal learns low-dimensional representations for device data that contain the least information about the private attribute in each device's data—thus protecting attribute privacy, and the other MI goal ensures the learnt representations include as much in-

Methods	Task-Agnostic	Efficient	Provable	Accurate
MPC			✓	✓
AT		✓		✓
MC	✓	✓		
DP	✓	✓	✓	
TAPPFL	✓	✓	✓	✓

Table 1: Comparisons of the PPFL methods.

formation as possible about the training data—thus maintaining FL utility. Our TAPPFL is task-agnostic as our formulation does not need to know the FL task. However, the true MI values are challenging to compute, due to that they deal with high-dimensional random variables and require to compute an intractable posterior distribution. Inspired by the MI neural estimators (Belghazi et al. 2018; Chen et al. 2016; Cheng et al. 2020), we recast calculating intractable exact MI values into deriving tractable (variational) MI bounds, where each variational bound is associated with a posterior distribution that can be parameterized via a neural network. Hence, estimating the true MI values reduces to training the parameterized neural networks. We further propose an alternative learning algorithm to train these neural networks and learn task-agnostic privacy-preserving representations for device data. We also derive privacy guarantees of TAPPFL against worst-case attribute inference attacks, as well as the inherent tradeoff between utility preservation and attribute privacy protection. Finally, we evaluate TAPPFL on multiple datasets and applications. Our results validate the learnt devices' data representations can be used to achieve high utility and maintain attribute privacy as well[1].

Our key contributions are highlighted as follows:

- **Algorithm:** We propose the first practical privacy-preserving FL mechanism (TAPPFL), i.e., task-agnostic, efficient, and accurate, against attribute inference attacks.
- **Theory:** TAPPFL has privacy guarantees and shows an inherent tradeoff between utility and privacy.
- **Evaluation:** TAPPFL is effective against attribute inference attacks and shows advantages over the baselines.

Related Work

Privacy-preserving FL against inference attacks. *Secure multi-party computation* (Danner and Jelasity 2015; Mohassel and Zhang 2017; Bonawitz et al. 2017; Melis et al. 2019), *adversarial training* (Oh, Fritz, and Schiele 2017; Wu et al. 2018; Madras et al. 2018; Pittaluga, Koppal, and Chakrabarti 2019; Liu et al. 2019; Kim et al. 2019), *model compression* (Zhu, Liu, and Han 2019), and *differential privacy (DP)* (Pathak, Rane, and Raj 2010; Shokri and Shmatikov 2015; Hamm, Cao, and Belkin 2016; McMahan et al. 2018; Geyer, Klein, and Nabi 2017; Wei et al. 2020) are the four typical privacy-preserving FL methods. For example, Bonawitz et al. (2017) design a secure multi-party aggregation for FL, where devices are required to encrypt their local models before uploading them to the server. However, it incurs an intolerable computational overhead and may need to know the specific FL task in advance. Adversarial training methods are inspired by GAN (Goodfellow et al. 2014). These methods adopt adversarial learning to learn obfuscated features from the training data so that their privacy information cannot be inferred from a learnt model. However, these methods also need to know the FL task and lack of formal privacy guarantees. Zhu, Liu, and Han (2019) apply gradient compression/sparsification to defend against privacy leakage from shared local models. However, to achieve a desirable privacy protection, such approaches require high compression rates, leading to intolerable utility losses. In addition, it does not have formal privacy guarantees. Shokri and Shmatikov (2015) propose a collaborative learning method where the sparse vector is adopted to achieve DP. However, DP methods have high utility losses.

Mutual information for fair representation learning. Moyer et al. (2018); Song et al. (2019) leverage MI to perform fair representation learning. The goal of fair representation learning is to encode the input data into a representation that aims to mitigate bias, e.g., demographic disparity, towards a private group/attribute. For instance, when achieving demographic parity, the predicted label should be independent of the private attribute. For instance, (Moyer et al. 2018) proposed to use the information bottleneck objective (Alemi et al. 2017) and train a variational auto-encoder (Kingma, Welling et al. 2019) to censor a private attribute by minimizing a variational bound on the MI between the learnt representations and the private attribute.

Besides the difference of the studied problem (fairness vs privacy-protection), other key differences are: these fair methods need to know the learning task in advance, and only estimate the MI for low-dimensional variables. In contrast, our method is task-agnostic and designs a novel MI estimator to handle the challenging high-dimensional variables. Further, these methods do not have theoretical fairness guarantees, while ours has formal privacy guarantees.

Mutual information for private representation learning. The most closely related to our work is DPFE (Osia et al. 2018), which also formulates learning private representations via MI objectives. However, there are major differences: i) DPFE knows the primary task, i.e., *task-specific*, while ours is task-agnostic; ii) DPFE does not have formal privacy guarantees, while ours do have; iii) DPFE estimates MI by making assumptions on distributions of the random variable, e.g., Gaussian. In contrast, our method derives MI bounds on random variables whose distributions can be *arbitrary*, and then trains networks to approximate the true MI.

Mutual information estimation. Accurately estimating MI between any random variables is challenging (Belghazi et al. 2018). Recent methods (Alemi et al. 2017; Belghazi et al. 2018; Oord, Li, and Vinyals 2018; Poole et al. 2019; Hjelm et al. 2019; Cheng et al. 2020; Wang et al. 2021) propose to first derive (upper or lower) MI bounds by introducing auxiliary variational distributions and then train parameterized neural networks to estimate variational distributions and approximate true MI. For instance, MINE (Belghazi et al. 2018) views MI as a KL divergence between the joint and marginal distributions, converts it into the dual representation, and derives a lower MI bound. Cheng et al. (2020) de-

[1]Source code and full version: https://github.com/TAPPFL

sign a Contrastive Log-ratio Upper Bound of MI, which connects MI with contrastive learning (Oord, Li, and Vinyals 2018), and estimates MI as the difference of conditional probabilities between positive and negative sample pairs.

Background and Problem Definition

Federated learning (FL). The FL paradigm enables a server to coordinate the training of multiple local devices through multiple rounds of global communications, without sharing the local data. Suppose there are M devices $\mathcal{C} = \{C_i\}_{i=1}^{M}$ and a server S participating in FL. Each device C_i has data samples $\mathbf{x}^i$ from a distribution $\mathcal{D}^i$ over the sample space $\mathcal{X}^i$. In each round t, each device C_i first downloads the previous round's global model (e.g., Θ_{t-1}) from the server, and then updates its local model (e.g., Θ_t^i) using the local data $\{\mathbf{x}^i\}$ and global model Θ_{t-1}. The server S then randomly collects a set of (e.g., K) current local models in devices (e.g., $\mathcal{C}_K$) and updates the global model for the next round using an aggregation algorithm. For example, when using the most common FedAvg (McMahan et al. 2017b), the server updates the global model as $\Theta_t \leftarrow \sum_{i\in\mathcal{C}_K} \frac{n_i}{\sum_{i\in\mathcal{C}_k} n_i}\Theta_t^i$, where n_i is the size of the training data of device C_i.

Threat model and problem definition. We assume each device C_i's data has its *private attribute* and denote it as u^i. Each device C_i aims to learn a feature extractor $f_{\Theta^i}: \mathcal{X}^i \rightarrow \mathcal{R}^i$, parameterized by Θ^i, that maps data samples from input space $\mathcal{X}^i$ to the latent representation space $\mathcal{R}^i$; and we denote the learnt representation for a sample $\mathbf{x}^i$ as $\mathbf{r}^i = f_{\Theta^i}(\mathbf{x}^i)$. The learnt representations can be used for downstream tasks, e.g., next-word-prediction on smart phones (Li et al. 2020b). We assume the server S is honest-but-curious and it has access to the learnt representations $\{\mathbf{r}^i\}$ of device data. The server's purpose is to infer any private attribute $\mathbf{u}^i$ through the representations without tampering the FL training process. Our goal is to learn the feature extractor f_{Θ^i} per device such that it protects the private attribute u^i from being inferred, as well as preserving the primary FL task utility. For a general purpose, we assume the primary FL task is unknown (i.e., task-agnostic) to the defender (i.e., who learns the feature extractor). W.l.o.g, the protected attribute is different from the primary task label.

Design of TAPPFL

In this section, we will design our task-agnostic privacy-preserving FL (TAPPFL) method against attribute inference attacks. Our TAPPFL is inspired by information theory. We show that our TAPPFL has provable privacy guarantees, as well as inherent utility-privacy tradeoffs.

Formulating TAPPFL via MI Objectives

For ease of description, we choose a device C_i and show how to learn its privacy-preserving feature extractor f_{Θ^i}. Our goal is to transform the data $\mathbf{x}^i \sim \mathcal{D}^i$ into a representation $\mathbf{r}^i = f_{\Theta^i}(\mathbf{x}^i)$ that satisfies the below two goals:

- **Goal 1: Privacy protection.** $\mathbf{r}^i$ contains as less information as possible about the private attribute u^i. Ideally, when $\mathbf{r}^i$ does not include information about u^i, it is impossible for the server to infer u^i from $\mathbf{r}^i$.
- **Goal 2: Utility preservation.** $\mathbf{r}^i$ includes as much information about the raw data $\mathbf{x}^i$ as possible. Ideally, when $\mathbf{r}^i$ retains the most information about $\mathbf{x}^i$, the model trained on $\mathbf{r}^i$ will have the same performance as the model trained on the raw $\mathbf{x}^i$, thus preserving utility. Note that this goal does not know the FL task, thus task-agnostic.

We propose to formalize the above two goals via MI objectives. In information theory, MI is a measure of shared information between two random variables, and offers a metric to quantify the "amount of information" obtained about one random variable by observing the other random variable. Formally, we quantify the privacy protection and utility reservation goals using two MI objectives as follows:

$$\textbf{Achieving Goal 1:} \quad \min_{\Theta^i} I(\mathbf{r}^i; u^i);$$
$$\textbf{Achieving Goal 2:} \quad \max_{\Theta^i} I(\mathbf{x}^i; \mathbf{r}^i | u^i). \tag{1}$$

where $I(\mathbf{r}^i; u^i)$ is the MI between $\mathbf{r}^i$ and u^i, and we minimize such MI to maximally reduce the correlation between $\mathbf{r}^i$ and u^i. $I(\mathbf{x}^i; \mathbf{r}^i | u^i)$ is the MI between $\mathbf{x}^i$ and $\mathbf{r}^i$ given u^i. We maximize such MI to keep the raw information in $\mathbf{x}^i$ as much as possible in $\mathbf{r}^i$ and remove the information that $\mathbf{x}^i$ contains about the private u^i to leak into $\mathbf{r}^i$.

Estimating MI via Tractable Variational Bounds

The key challenge of solving the above two MI objectives is that calculating an MI between two arbitrary random variables is likely to be infeasible (Peng et al. 2018). To address it, we are inspired by the existing MI neural estimation methods (Alemi et al. 2017; Belghazi et al. 2018; Oord, Li, and Vinyals 2018; Poole et al. 2019; Hjelm et al. 2019; Cheng et al. 2020), which convert the intractable exact MI calculations to the tractable variational MI bounds. Specifically, we first obtain a MI upper bound for privacy protection and a MI lower bound for utility preserving via introducing two auxiliary posterior distributions, respectively. Then, we parameterize each auxiliary distribution with a neural network, and approximate the true posteriors by minimizing the upper bound and maximizing the lower bound through training the involved neural networks.

Minimizing upper bound MI for privacy protection. We propose to adapt the variational upper bound CLUB proposed in (Cheng et al. 2020) to bound $I(\mathbf{r}^i; u^i)$. Specifically,

$$I(\mathbf{r}^i; u^i) \leq I_{vCLUB}(\mathbf{r}^i; u^i) \tag{2}$$
$$= \mathbb{E}_{p(\mathbf{r}^i, u^i)}[\log q_{\Psi^i}(u^i|\mathbf{r}^i)] - \mathbb{E}_{p(\mathbf{r}^i)p(u^i)}[\log q_{\Psi^i}(u^i|\mathbf{r}^i)],$$

where $q_{\Psi^i}(u^i|\mathbf{r}^i)$ is an auxiliary posterior distribution of $p(u^i|\mathbf{r}^i)$ needing to satisfy the condition:

$$KL(p(\mathbf{r}^i, u^i)||q_{\Psi^i}(\mathbf{r}^i, u^i)) \leq KL(p(\mathbf{r}^i)p(u^i)||q_{\Psi^i}(\mathbf{r}^i, u^i)),$$

where $KL[q(\cdot)||p(\cdot)]$ is the Kullback-Leibler divergence between two distributions $q(\cdot)$ and $p(\cdot)$ and is nonnegative. To achieve this, we thus need to minimize:

$$\begin{aligned}
&\min_{\Psi^i} KL(p(\mathbf{r}^i, u^i)||q_{\Psi^i}(\mathbf{r}^i, u^i)) \\
=&\min_{\Psi^i} KL(p(u^i|\mathbf{r}^i)||q_{\Psi^i}(u^i|\mathbf{r}^i)) \\
=&\min_{\Psi^i} \mathbb{E}_{p(\mathbf{r}^i,u^i)}[\log p(u^i|\mathbf{r}^i)] - \mathbb{E}_{p(\mathbf{r}^i,u^i)}[\log q_{\Psi^i}(u^i|\mathbf{r}^i))] \\
\Leftrightarrow&\max_{\Psi^i} \mathbb{E}_{p(\mathbf{r}^i,u^i)}[\log q_{\Psi^i}(u^i|\mathbf{r}^i)],
\end{aligned} \tag{3}$$

where we use that the first term $\mathbb{E}_{p(\mathbf{r}^i,u^i)}[\log p(u^i|\mathbf{r}^i)]$ in the second-to-last Equation is irrelevant to Ψ^i.

Finally, our **Goal 1** for privacy protection is reformulated as solving the below min-max objective function:

$$\min_{\Theta^i} I(\mathbf{r}^i;u^i) = \min_{\Theta^i}\min_{\Psi^i} I_{vCLUB}(\mathbf{r}^i;u^i)$$
$$\Longleftrightarrow \min_{\Theta^i}\max_{\Psi^i} \mathbb{E}_{p(\mathbf{r}^i,u^i)}[\log q_{\Psi^i}(u^i|\mathbf{r}^i)]. \quad (4)$$

Remark. Equation (4) can be interpreted as an *adversarial game* between: (1) an adversary q_{Ψ^i} (i.e., attribute inference classifier) who aims to infer the private attribute u^i from $\mathbf{r}^i$; and (2) a defender (i.e., the feature extractor f_{Θ^i}) who aims to protect u^i from being inferred from $\mathbf{r}^i$.

Maximizing lower bound MI for utility preservation. We derive the lower bound of the MI $I(\mathbf{x}^i;\mathbf{r}^i|u^i)$ as follows:

$$\begin{aligned} I(\mathbf{x}^i;\mathbf{r}^i|u^i) &= H(\mathbf{x}^i|u^i) - H(\mathbf{x}^i|\mathbf{r}^i,u^i) \\ &= H(\mathbf{x}^i|u^i) + \mathbb{E}_{p(\mathbf{x}^i,\mathbf{r}^i,u)}[\log p(\mathbf{x}^i|\mathbf{r}^i,u^i))] \\ &= H(\mathbf{x}^i|u^i) + \mathbb{E}_{p(\mathbf{x}^i,\mathbf{r}^i,u)}[\log q_{\Omega^i}(\mathbf{x}^i|\mathbf{r}^i,u^i))] \\ &\quad + \mathbb{E}_{p(\mathbf{x}^i,\mathbf{r}^i,u)}[KL(p(\cdot|\mathbf{r}^i,u^i)||q_{\Omega^i}(\cdot|\mathbf{r}^i,u^i))] \\ &\geq H(\mathbf{x}^i|u^i) + \mathbb{E}_{p(\mathbf{x}^i,\mathbf{r}^i,u)}[\log q_{\Omega^i}(\mathbf{x}^i|\mathbf{r}^i,u^i))], \end{aligned} \quad (5)$$

where q_{Ω^i} is an arbitrary auxiliary posterior distribution that aims to maintain the information in $\mathbf{x}^i$, and $H(\mathbf{x}^i|u^i)$ is the condition entropy that is constant. Hence, our **Goal 2** can be rewritten as the following max-max objective function:

$$\max_{\Theta^i} I(\mathbf{x}^i;\mathbf{r}^i|u^i) \Longleftrightarrow \max_{\Theta^i}\max_{\Omega^i} \mathop{\mathbb{E}}_{p(\mathbf{x}^i,\mathbf{r}^i,u^i)}\left[\log q_{\Omega^i}[(\mathbf{x}^i|\mathbf{r}^i,u^i)\right]. \quad (6)$$

Remark. Equation (6) can be interpreted as a *cooperative game* between the feature extractor f_{Θ^i} and q_{Ω^i} who aim to preserve the utility collaboratively.

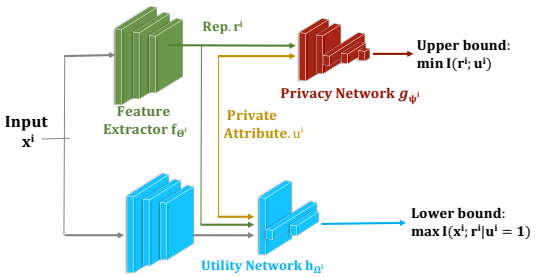
Objective function of TAPPFL. By combining Equations (4) and (6) and considering all devices, our final objective function of learning the task-agnostic privacy-preserving representations in FL is as follows:

$$\sum_{C_i\in\mathcal{C}} \max_{\Theta^i}\Big(\lambda_i \min_{\Psi^i} -\mathbb{E}_{p(u^i,\mathbf{x}^i)}\left[\log q_{\Psi^i}(u^i|f_{\Theta^i}(\mathbf{x}^i))\right]$$
$$+(1-\lambda_i)\max_{\Omega^i}\mathbb{E}_{p(\mathbf{x}^i,u^i)}\left[\log q_{\Omega^i}[(\mathbf{x}^i|f_{\Theta^i}(\mathbf{x}^i),u^i)\right]\Big), \quad (7)$$

where $\lambda_i \in [0,1]$ achieves a tradeoff between privacy and utility for device C_i. That is, a larger λ_i indicates a stronger attribute privacy protection for C_i's data, while a smaller λ_i indicates a better utility preservation for C_i's data.

Implementation in Practice

Directly calculating the exact expectation in Equation (7) is challenging. In practice, Equation (7) can be solved via training three parameterized neural networks, i.e., the feature extractor f_{Θ^i}, the privacy-protection network g_{Ψ^i} associated with the auxiliary distribution q_{Ψ^i}, and the utility-preservation network h_{Ω^i} associated with the auxiliary distribution q_{Ω^i}, using sampled data from each device C_i. Specifically, in each device C_i, we first collect a set of samples $\{\mathbf{x}_j^i\}$ and the associated private attributes $\{u_j^i\}$ from $\mathcal{D}^i$. Note that, as our TAPPFL is task-agnostic, we do not know the sample labels for the FL task. With it, we can then approximate the expectation terms in Equation (7).



(a) TAPPFL device training

Figure 1: TAPPFL device training.

Specifically, we approximate the first expectation term as

$$\mathbb{E}_{p(u^i,\mathbf{x}^i)}\log q_{\Psi^i}(u^i|f_{\Theta^i}(\mathbf{x}^i)) \approx -\sum\nolimits_j CE(u_j^i, g_{\Psi^i}(f_{\Theta^i}(\mathbf{x}_j^i))),$$

where $CE(\cdot)$ means the cross-entropy error function.

Moreover, note that the data $\mathbf{x}$ and the representation $\mathbf{r}$ are rather high-dimensional. To address it, we propose to use the *Jensen-Shannon* divergence (JSD) (Hjelm et al. 2019) for high-dimensional MI estimation and approximate the second expectation term associated with q_{Ω^i} as below:

$$\begin{aligned} &\mathbb{E}_{p(\mathbf{x}^i,u^i)}\log q_{\Omega^i}(\mathbf{x}^i|f_{\Theta^i}(\mathbf{x}_i),u^i) = I^{(JSD)}_{\Theta^i,\Omega^i}(\mathbf{x}^i;f_{\Theta^i}(\mathbf{x}^i),u^i) \\ &= \mathbb{E}_{(\mathbf{x}^i,u^i)}[-\text{sp}(-h_{\Omega^i}(\mathbf{x}^i,f_{\Theta^i}(\mathbf{x}^i),u^i)] \\ &\quad - \mathbb{E}_{(\mathbf{x}^i,u^i,\bar{\mathbf{x}}^i)}[\text{sp}(h_{\Omega^i}(\bar{\mathbf{x}}^i,f_{\Theta}(\mathbf{x}^i),u^i)], \end{aligned}$$

where $\text{sp}(z) = \log(1+\exp(z))$ is the softplus function, $\bar{\mathbf{x}}^i$ is an independent sample from the same distribution as $\mathbf{x}^i$, and the expectation can be replaced by samples $\{\mathbf{x}_j^i, \bar{\mathbf{x}}_j^i, u_j^i\}$.

Figure 1 illustrates our TAPPFL. It needs to simultaneously train three neural networks, i.e., the feature extractor f_{Θ^i}, the privacy-protection network g_{Ψ^i}, and the utility-preservation network h_{Ω^i}, in each device C_i. In particular, the server first initializes a global model Θ_0 and broadcasts Θ_0 to all devices; and the devices initializes $\{\Psi_0^i\}$ and $\{\Omega_0^i\}$ locally. Then the training procedure involves two iterative steps. For example, in the t-th round: In Step I, each device updates Θ_t^i using the received Θ_{t-1} from the server, and locally updates Ψ_t^i and Ω_t^i using Ψ_{t-1}^i and Ω_{t-1}^i based on its training data; and the devices send the updated $\{\Theta_t^i\}$ to the server. In Step II, the server selects a set of $\{\Theta_t^i\}$ and updates the global model Θ_t by aggregating these models via, e.g., Fedvg (McMahan et al. 2017b), and broadcasts Θ_t to all devices. We repeat the two steps alternately until convergence or reaching the maximum number of iterations. Algorithm 1 in the full version details the TAPPFL training process.

Theoretical Results

Inherent utility-privacy tradeoff. We consider the attribute has a binary value and the primary FL task is binary classification. We will leave it as a future work to generalize our results to multi-value attributes and multi-class classification.

CIFAR10				Loans			Adult income			Adult income		
Private attribute: Animal or not (binary)				Race (binary)			Gender (binary)			Marital status (7 values)		
λ	Test Acc	Infer. Acc	Gap	Test Acc	Infer. Acc	Gap	Test Acc	Infer. Acc	Gap	Test Acc	Infer. Acc	Gap
0	0.89	0.74	0.24	0.98	0.74	0.24	0.825	0.700	0.20	0.825	0.375	0.232
0.25	0.88	0.64	0.14	0.93	0.72	0.22	0.750	0.550	0.05	0.800	0.275	0.112
0.5	0.76	0.60	0.10	0.88	0.70	0.20	0.750	0.550	0.05	0.800	0.250	0.107
0.75	0.67	0.56	0.06	0.84	0.63	0.13	0.725	0.550	0.05	0.725	0.243	0.043
1	0.58	0.52	0.02	0.82	0.57	0.07	0.700	0.525	0.025	0.700	0.175	0.032

Table 2: Test accuracy (Test Acc.) vs. attribute inference accuracy (Infer. Acc.) on the considered three datasets.

Let $\mathcal{A}$ be the set of all binary attribute inference classifiers, i.e., $\mathcal{A} = \{A : \mathbf{r}^i \in \mathcal{R}^i \rightarrow \{0,1\}, \forall C_i\}$. Let $\mathcal{D}^i$ be a joint distribution over the input $\mathbf{x}^i$, sensitive attribute u^i, and label $\mathbf{y}^i$ for device C_i. W.l.o.g, we assume the representation space is bounded, i.e., $\max_{C_i \in \mathcal{C}} \max_{\mathbf{r}^i \in \mathcal{R}^i} \|\mathbf{r}^i\| \leq R$. Moreover, we denote the binary task classifier as $c : \mathbf{r}^i \rightarrow \{0,1\}$, which predicts data labels based on the learnt representation. We further define the *advantage* of the worst-case adversary with respect to the joint distribution $\mathcal{D}^i$ as below:

$$\text{Adv}_{\mathcal{D}^i}(\mathcal{A}) = \max_{A \in \mathcal{A}} |\Pr_{\mathcal{D}^i}(A(\mathbf{r}^i) = a | u^i = a) - \Pr_{\mathcal{D}^i}(A(\mathbf{r}^i) = a | u^i = 1 - a)|, \forall a = \{0,1\}. \quad (8)$$

If $\text{Adv}_{\mathcal{D}^i}(\mathcal{A}) = 1$, this means an adversary can *completely* infer the privacy attribute through the learnt representations. In contrast, $\text{Adv}_{\mathcal{D}^i}(\mathcal{A}) = 0$ means an adversary obtains a *random guessing* inference performance. Our goal is thus to learn the representations such that $\text{Adv}_{\mathcal{D}^i}(\mathcal{A})$ is small. The proofs are in the full version: https://github.com/TAPPFL.

Theorem 1. *Let $\mathbf{r}^i$ be the representation with a bounded norm R (i.e., $\max_{\mathbf{r}^i \in \mathcal{R}^i} \|\mathbf{r}^i\| \leq R$) learnt by the feature extractor f_{Θ_i} for device C_i's data $\mathbf{x}^i$, and $\mathcal{A}$ be the set of all binary attribute inference classifiers. Assume the task classifier c is C_L-Lipschitz, i.e., $\|c\|_L \leq C_L$. Then, each C_i's classification error err_i can be bounded as below:*

$$err_i \geq \Delta_{\mathbf{y}^i|u^i} - 2R \cdot C_L \cdot Adv_{\mathcal{D}^i}(\mathcal{A}), \quad (9)$$

where $\Delta_{y^i|u^i} = |Pr_{\mathcal{D}^i}(y^i = 1 | u^i = 0) - Pr_{\mathcal{D}^i}(y^i = 1 | u^i = 1)|$ is a device-dependent constant.

Remark. Theorem 1 states that, for a device-dependent constant $\Delta_{\mathbf{y}^i|u^i}$, any primary task classifier using representations learnt by the feature extractor f_{Θ_i} has to incurs a classification error on at least a private attribute—The smaller/larger the advantage $\text{Adv}_{\mathcal{D}_i}(\mathcal{A})$ is, the larger/smaller the lower error bound. Note that the lower bound is independent of the adversary, meaning it covers the *worst-case* adversary. Thus, Equation (9) reflects an inherent trade-off between utility preservation and attribute privacy leakage.

Privacy guarantees against attribute inference. The attribute inference accuracy incurred by the worst-case adversary is bounded in the following theorem:

Theorem 2. *Let Θ^i_* (resp. $\mathbf{r}^i_*$) be the learnt optimal feature extractor parameters (resp. optimal representations) by Equation (7) for device C_i's data. Define $H^i_* = H(u^i|\mathbf{r}^i_*)$. Then, for any attribute inference adversary $\mathcal{A} = \{A : \mathbf{r}^i \rightarrow u^i\}$, $Pr(A(\mathbf{r}^i_*) = u^i) \leq 1 - \frac{H^i_*}{2\log_2(6/H^*_i)}$.*

Remark. Theorem 2 shows that when the conditional entropy $H^i_* = H(u^i|\mathbf{r}^i_*)$ is larger, the attribute inference accuracy induced by any adversary is smaller, i.e., the less attribute privacy is leaked. From another perspective, as $H(u^i|\mathbf{r}^i_*) = H(u^i) - I(u^i; \mathbf{r}^i_*)$, achieving the largest $H(u^i|\mathbf{r}^i_*)$ indicates minimizing the mutual information $I(u^i; \mathbf{r}^i_*)$—This is exactly our **Goal 1** aims to achieve.

Experiments

Experimental Setup

Datasets and applications. We evaluate our TAPPFL using three datasets from different applications. CIFAR-10 (Krizhevsky 2009) is an image dataset. The primary task is to predict the label of the image, while the private attribute is a binary attribute indicating if an image belongs to an animal or not. For the Loans dataset (Hardt, Price, and Srebro 2016), the primary task is to predict the affordability of the person asking for the loan while protecting their race. For the Adult income dataset (Becker and Kohavi 1996), predicting whether the income of a person is above \$50K or not is the primary task. The private attributes are the gender and the marital status. More detailed descriptions of these datasets and the training/testing sets can be found in the full version.

Parameter settings. We use a total of 100 devices participating in FL training. By default, the server randomly selects 10% devices and uses FedAvg (McMahan et al. 2017b) to aggregate devices' feature extractor parameters in each round. In each device, we train the three parameterized neural networks via the Stochastic Gradient Descent (SGD) algorithm, where we set the local batch size to be 10 and use 10 local epochs, and the learning rate in SGD is 0.01. A detailed architecture of each neural network can be found in Table 3 in the full version. Before the learning, we first pretrain the feature extractor network only to obtain a good initialization, i.e., high utility. The number of global rounds is set to be 20. In TAPPFL, for simplicity, we set $\lambda_i = \lambda$ for all devices and all devices share the same private attribute. The TAPPFL algorithm is implemented in PyTorch. We use the Chameleon Cloud platform offered by the NSF (Keahey et al. 2020) (CentOS7-CUDA 11 with Nvidia Rtx 6000).

Evaluation metrics. We evaluate TAPPFL in terms of both utility preservation and privacy protection. We use the testing accuracy (i.e., device's feature extractor + utility network on the primary task's test set) to measure utility preservation; and attribute inference accuracy (i.e., device's feature extractor + privacy network on the privacy task's test set) to measure the privacy leakage. The larger testing accuracy,

the better utility preservation; the inference accuracy closer to random guessing, the less attribute privacy leakage.

Experimental Results

Utility-privacy tradeoff. According to Equation (7), when $\lambda = 0$ the first term of the objective function is disregarded, meaning that the protection of the private attribute is not considered. On the contrary, the second term is disappeared when $\lambda = 1$, or in other words, we only consider protecting the private attribute and utility is not preserved. Our goal is to achieve a better trade-off by tuning λ within $[0, 1]$, which allows preserving the FL utility and protecting the attribute privacy at the same time. Table 2 shows the testing accuracy and average attribute inference accuracy of all devices in the considered datasets, where we set five different λ values, i.e., 0, 0.25, 0.5, 0.75, and 1.0. We also show the gap between the attribute inference accuracy and the random guessing. The smaller the gap, the better the privacy protection. Ideally, when there is no gap, the learnt representation by our TAPPFL does not allow the adversary (i.e., the server) to infer *any* information related to the private attribute. Specifically, we have the following observations: 1) The testing accuracy is the largest when $\lambda = 0$, hence the utility is maintained the most. However, the attribute inference accuracy is also the highest, indicating leaking the most attribute privacy. 2) The attribute inference accuracy is the closest to random guessing when $\lambda = 1$, meaning the attribute privacy is protected the most. However, the testing accuracy is also the smallest, indicating the utility is not well maintained. 3) When $0 < \lambda < 1$, our TAPPFL achieves both reasonable testing accuracy and attribute inference accuracy—This indicates TAPPFL has a better utility-privacy tradeoff. Note that our TAPPFL does not know primary task's labels and learns the task-agnostic representations for device data during the entire training.

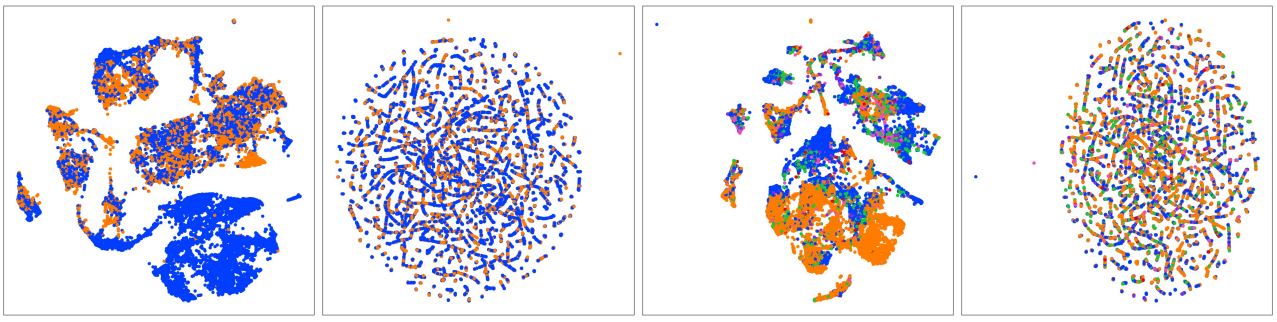
Mutual information vs. tradeoff parameter λ. Further, we analyze our TAPPFL via plotting the two MI scores (i.e., the CE loss associated with **Goal 1 (privacy protection)** and JSD loss associated with **Goal 2 (utility preservation)**) vs. λ. Note that the CE and JSD loss are inversely proportional to the MI in the two goals. Figure 4 in the full version shows the results. Each point indicates either a CE loss or JSD loss at a selected λ. The tendency of these scores in function of λ is presented by a trend line, which is computed using a least squares polynomial fit of first degree. We observe that: 1) When the trade-off parameter λ is low, the privacy protection is not carefully considered, which is translated into a high MI between the learnt representation and the private attribute, thus the CE loss is relatively small. On the other hand, the utility is well-preserved, resulting also into a high MI between the input data and the learnt representation conditioned on the private attribute. So the JSD loss is relatively small. 2) Contrarily, for high values of λ, the privacy is largely protected in exchange for a large utility loss. Specifically, as λ increases, the CE between the private attribute and the learnt representation increases, which is translated into a decrease of their MI, thus better protecting attribute privacy. Though not easily appreciated in the curves, the JSD loss tends to increase, thus reducing the utility.

Visualization of the learnt representations. In this experiment, we leverage the t-SNE embedding algorithm (Van der Maaten and Hinton 2008) to visualize the learnt representations by our trained feature extractor for the device data, and those without our feature extractor. λ is chosen in Table 2 that achieves the best utility-privacy tradeoff. Figure 2 shows the 2D T-SNE visualization results, where each color corresponds to a private attribute value. We can observe that the 2D T-SNE embeddings of the raw input data form some clusters for the private attributes, meaning the private attributes can be easily inferred, e.g., the t-SNE representations via training a multi-class classifier. On the contrary, the 2D T-SNE embeddings of the learnt representations by our TAPPFL for different attribute values are completely mixed, which thus makes it difficult for a malicious server to infer the private attributes from the learnt representations.

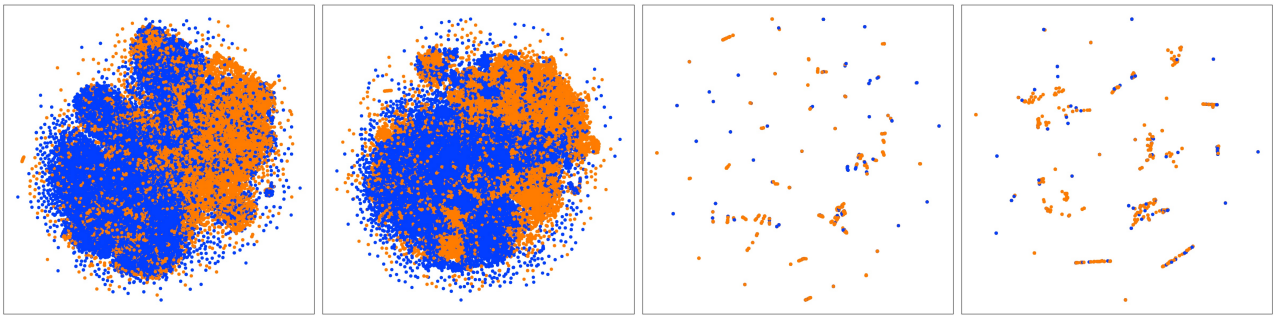
Comparison with task-agnostic defenses. In this experiment, we compare our TAPPFL with the two *task-agnostic* privacy protection methods, i.e., differential privacy (DP) (Wei et al. 2020) and model compression (MC) (Zhu, Liu, and Han 2019) (See Table 1). MC prunes the devices' feature extractor parameters whose magnitude are smaller than a threshold, and the devices only share parameters larger than the threshold to the server. DP ensures provable privacy guarantees (Wang et al. 2022). Specifically, DP randomly injects noise into the feature extractor's parameters and uploads the noisy parameters to the server. The server then performs the aggregation using the noisy parameters. Here, we consider applying the Gaussian noise and Laplacian noise to develop two DP baselines (Mohammady et al. 2020), i.e., DP-Gaussian and DP-Laplace (note that the DP protection in (Wei et al. 2020) is very weak due to very high ϵ such as 50 and 100). Thus, we tune the hyperparameter, i.e., noise variance in DP and pruning rate in MC, such that DP and MC have the same attribute inference accuracy as TAPPFL, and then compare their utility (i.e., testing accuracy). Figure 3 shows the comparison results on CIFAR10, where we set five attribute inference accuracy as 0.55, 0.60, 0.65, 0.70, and 0.75, respectively. We can observe that TAPPFL is significantly better than them. *Note that we also evaluate DP by injecting noises to the final/input layer (as other existing methods did) and observe that DP-Gaussian and DP-Laplacian have very close (bad) performance as those injecting noises into the feature extractor's parameters. We do not show these results for simplicity.*

Discussion

Comparison with task-specific defenses. We also compare TAPPFL with state-of-the-art task-specific defenses, i.e., DPFE (Osia et al. 2018) and adversarial training (AT) (Li et al. 2020a) and show results on Loans under the same setting. *Note that DPFE and AT know the primary task.* We have the following observations: 1) Comparing with DPFE, with a same test accuracy as 0.84, TAPPFL's attribute inference accuracy is 0.63, while DPFE's is 0.77—showing a much worse protection than TAPPFL. One key reason is that DPFE estimates MI by assuming random variables to be Gaussian, which is inaccurate. Instead, TAPPFL does not have any assumption on the distributions of the random



(a) Adult (G.): Raw input (b) Adult (G.): Learnt rep. (c) Adult (M.): Raw input (d) Adult (M.): Learnt rep.



(e) CIFAR10: Raw input (f) CIFAR10: Learnt rep. (g) Loans: Raw input (h) Loans: Learnt rep.

Figure 2: 2D tSNE of learnt representations by TAPPFL and of the raw input on a random client. Color means attribute value.

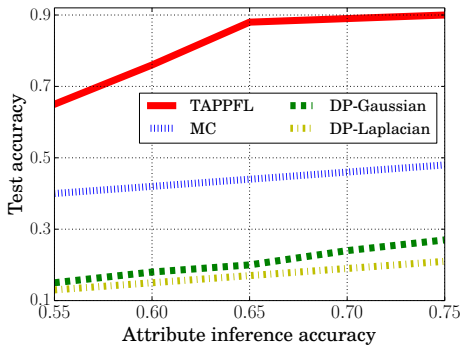


Figure 3: Compared defense results on CIFAR10.

variables, meaning it can handle *arbitrary* random variable distributions. 2) Comparing with AT, our TAPPFL's best-tradeoff is (test accuracy, inference accuracy) = (0.84, 0.63), while AT's is (0.86, 0.63)—slightly better than ours. This is because both TAPPFL and AT involve adversarial training, but AT uses primary *task labels*, while ours does not.

Defending against the state-of-the-art de-censoring representation attacks. Song and Shmatikov (2020) designed a de-censoring representation attack to better infer private attributes. Here, we test our TAPPFL against this attack on Loans. Specifically, by applying TAPPFL, when the test accuracy is 0.84, the inference accuracy *without* Song and Shmatikov (2020) is 0.63, while *with* Song and Shmatikov (2020) is 0.73, showing its better attack effectiveness. We then enhance privacy protection until the inference accuracy without Song and Shmatikov (2020) is 0.57. In this case, its inference accuracy reduces to 0.60. The results show Song and Shmatikov (2020)'s performance could be largely reduced with more privacy protection imposed by TAPPFL. The fundamental reason is that TAPPFL has privacy guarantees against the (worst-case) inference attack.

Conclusion

We study privacy-preserving federated learning (FL) against the attribute inference attack, i.e., an honest-but-curious server infers sensitive information in the device data from shared device models. To this end, we design a task-agnostic and provably privacy-preserving representation learning framework for FL (termed TAPPFL) from the information-theoretic perspective. TAPPFL is formulated via two mutual information goals: one goal learns low-dimensional representations for device data that contain the least information about the data's private attribute, and the other one includes as much information as possible about the raw data, in order to maintain FL utility. TAPPFL also has upper bounded privacy leakage of the private attributes. Extensive results on various datasets from different applications show that TAPPFL can well protect the attributes (i.e., attribute inference accuracy is close to random guessing), and obtains a high utility. TAPPFL is also shown to largely outperform the state-of-the-art task-agnostic defenses.

Acknowledgements

We thank the anonymous reviewers for their constructive feedback. Wang is partially supported by the Cisco Research Award, National Science Foundation under grant Nos. ECCS-2216926 and CNS-2241713. Hong is partially supported by the National Science Foundation under grant Nos. CNS-2302689 and CNS-2308730. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies.

References

Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2017. Deep Variational Information Bottleneck. In *ICLR*.

Alibaba Federated Learning. 2023. FederatedScope. https://federatedscope.io/. Accessed: 2023-01-01.

Aono, Y.; Hayashi, T.; Wang, L.; and Moriai, S. 2017. Privacy-preserving deep learning: Revisited and enhanced. In *ATIS*.

Becker, B.; and Kohavi, R. 1996. Adult. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20.

Belghazi, M. I.; Baratin, A.; Rajeshwar, S.; Ozair, S.; Bengio, Y.; Courville, A.; and Hjelm, D. 2018. Mutual information neural estimation. In *ICML*.

Bonawitz, K.; Eichner, H.; Grieskamp, W.; and et al. 2019. Towards federated learning at scale: System design. *MLSys*.

Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H. B.; Patel, S.; Ramage, D.; Segal, A.; and Seth, K. 2017. Practical secure aggregation for privacy-preserving machine learning. In *CCS*.

Chen, X.; Duan, Y.; Houthooft, R.; Schulman, J.; Sutskever, I.; and Abbeel, P. 2016. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*.

Cheng, P.; Hao, W.; Dai, S.; Liu, J.; Gan, Z.; and Carin, L. 2020. CLUB: A Contrastive Log-ratio Upper Bound of Mutual Information. In *ICML*.

Dang, T.; Thakkar, O.; Ramaswamy, S.; Mathews, R.; Chin, P.; and Beaufays, F. 2021. Revealing and protecting labels in distributed training. In *NeurIPS*.

Danner, G.; and Jelasity, M. 2015. Fully distributed privacy preserving mini-batch gradient descent learning. In *IFIP DAIS*.

Ganju, K.; Wang, Q.; Yang, W.; Gunter, C. A.; and Borisov, N. 2018. Property inference attacks on fully connected neural networks using permutation invariant representations. In *CCS*.

Geyer, R. C.; Klein, T.; and Nabi, M. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*.

Google Federated Learning. 2022. Federated Learning. https://federated.withgoogle.com/. Accessed: 2022-01-01.

Hamm, J.; Cao, Y.; and Belkin, M. 2016. Learning privately from multiparty data. In *ICML*.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. In *NIPS*.

Hjelm, R. D.; Fedorov, A.; Lavoie-Marchildon, S.; Grewal, K.; Bachman, P.; Trischler, A.; and Bengio, Y. 2019. Learning deep representations by mutual information estimation and maximization. In *ICLR*.

IBM Federated Learning. 2022. IBM Federated Learning. https://www.ibm.com/docs/en/cloud-paks/cp-data/4.0?topic=models-federated-learning. Accessed: 2022-11-21.

Jia, J.; Wang, B.; Zhang, L.; and Gong, N. Z. 2017. AttriInfer: Inferring user attributes in online social networks using markov random fields. In *WWW*.

Kaissis, G. A.; Makowski, M. R.; Rückert, D.; and Braren, R. F. 2020. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*.

Keahey, K.; Anderson, J.; Zhen, Z.; Riteau, P.; Ruth, P.; Stanzione, D.; Cevik, M.; Colleran, J.; Gunawi, H. S.; Hammock, C.; Mambretti, J.; Barnes, A.; Halbach, F.; Rocha, A.; and Stubbs, J. 2020. Lessons Learned from the Chameleon Testbed. In *USENIX ATC*.

Kim, T.-h.; Kang, D.; Pulli, K.; and Choi, J. 2019. Training with the invisibles: Obfuscating images to share safely for learning visual recognition models. *arXiv preprint arXiv:1901.00098*.

Kingma, D. P.; Welling, M.; et al. 2019. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4): 307–392.

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. *Master's thesis, University of Toronto*.

Li, A.; Guo, J.; Yang, H.; and Chen, Y. 2020a. Deepobfuscator: Adversarial training framework for privacy-preserving image classification. *arXiv preprint arXiv:1909.04126*.

Li, T.; Sahu, A. K.; Talwalkar, A.; and Smith, V. 2020b. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*.

Liu, S.; Du, J.; Shrivastava, A.; and Zhong, L. 2019. Privacy Adversarial Network: Representation Learning for Mobile Data Privacy. *IMWUT*, 3(4): 1–18.

Madras, D.; Creager, E.; Pitassi, T.; and Zemel, R. 2018. Learning adversarially fair and transferable representations. In *ICML*.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017a. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *AISTATS*.

McMahan, H. B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017b. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *AISTATS*.

McMahan, H. B.; Ramage, D.; Talwar, K.; and Zhang, L. 2018. Learning Differentially Private Recurrent Language Models. In *ICLR*.

Melis, L.; Song, C.; De Cristofaro, E.; and Shmatikov, V. 2019. Exploiting unintended feature leakage in collaborative learning. In *IEEE SP*.

Microsoft Federated Learning. 2022. FLUTE: A scalable federated learning simulation platform. https://www.microsoft.com/en-us/research/blog/flute-a-scalable-federated-learning-simulation-platform/. Accessed: 2022-05-16.

Mohammady, M.; Xie, S.; Hong, Y.; Zhang, M.; Wang, L.; Pourzandi, M.; and Debbabi, M. 2020. R2DP: A Universal and Automated Approach to Optimizing the Randomization Mechanisms of Differential Privacy for Utility Metrics with No Known Optimal Distributions. In *CCS*.

Mohassel, P.; and Zhang, Y. 2017. Secureml: A system for scalable privacy-preserving machine learning. In *IEEE SP*.

Moyer, D.; Gao, S.; Brekelmans, R.; Galstyan, A.; and Ver Steeg, G. 2018. Invariant representations without adversarial training. In *NeurIPS*.

Oh, S. J.; Fritz, M.; and Schiele, B. 2017. Adversarial image perturbation for privacy protection a game theory perspective. In *ICCV*.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Osia, S. A.; Taheri, A.; Shamsabadi, A. S.; Katevas, K.; Haddadi, H.; and Rabiee, H. R. 2018. Deep private-feature extraction. *IEEE TKDE*.

Pathak, M.; Rane, S.; and Raj, B. 2010. Multiparty differential privacy via aggregation of locally trained classifiers. In *NIPS*.

Peng, X. B.; Kanazawa, A.; Toyer, S.; Abbeel, P.; and Levine, S. 2018. Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow. *arXiv preprint arXiv:1810.00821*.

Pittaluga, F.; Koppal, S.; and Chakrabarti, A. 2019. Learning privacy preserving encodings through adversarial training. In *WACV*.

Poole, B.; Ozair, S.; Oord, A. v. d.; Alemi, A. A.; and Tucker, G. 2019. On variational bounds of mutual information. In *ICML*.

Rieke, N.; Hancox, J.; Li, W.; Milletari, F.; Roth, H. R.; Albarqouni, S.; Bakas, S.; Galtier, M. N.; Landman, B. A.; Maier-Hein, K.; et al. 2020. The future of digital health with federated learning. *NPJ digital medicine*, 3(1): 119.

Shokri, R.; and Shmatikov, V. 2015. Privacy-preserving deep learning. In *CCS*.

Song, C.; and Shmatikov, V. 2020. Overlearning Reveals Sensitive Attributes. In *ICLR*.

Song, J.; Kalluri, P.; Grover, A.; Zhao, S.; and Ermon, S. 2019. Learning controllable fair representations. In *AISTATS*.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *JMLR*.

Wainakh, A.; Ventola, F.; Müßig, T.; Keim, J.; Cordero, C. G.; Zimmer, E.; Grube, T.; Kersting, K.; and Mühlhäuser, M. 2022. User-Level Label Leakage from Gradients in Federated Learning. In *PTES*.

Wang, B.; Guo, J.; Li, A.; Chen, Y.; and Li, H. 2021. Privacy-Preserving Representation Learning on Graphs: A Mutual Information Perspective. In *KDD*.

Wang, H.; Sharma, J.; Feng, S.; Shu, K.; and Hong, Y. 2022. A Model-Agnostic Approach to Differentially Private Topic Mining. In *KDD*.

Wei, K.; Li, J.; Ding, M.; Ma, C.; Yang, H. H.; Farokhi, F.; Jin, S.; Quek, T. Q.; and Poor, H. V. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE TIFS*.

Wu, Z.; Wang, Z.; Wang, Z.; and Jin, H. 2018. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *ECCV*.

Zhu, L.; Liu, Z.; and Han, S. 2019. Deep leakage from gradients. In *NeurIPS*.