Privacy and Fairness in Federated Learning: On the Perspective of Tradeoff

HUIQIANG CHEN, TIANQING ZHU, and TAO ZHANG, University of Technology Sydney, Australia
WANLEI ZHOU, City University of Macau, China
PHILIP S. YU, University of Illinois at Chicago, US

Federated learning (FL) has been a hot topic in recent years. Ever since it was introduced, researchers have endeavored to devise FL systems that protect privacy or ensure fair results, with most research focusing on one or the other. As two crucial ethical notions, the interactions between privacy and fairness are comparatively less studied. However, since privacy and fairness compete, considering each in isolation will inevitably come at the cost of the other. To provide a broad view of these two critical topics, we presented a detailed literature review of privacy and fairness issues, highlighting unique challenges posed by FL and solutions in federated settings. We further systematically surveyed different interactions between privacy and fairness, trying to reveal how privacy and fairness could affect each other and point out new research directions in fair and private FL.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Computing methodologies** → *Machine learning;*

Additional Key Words and Phrases: Federated learning, data privacy, model fairness

ACM Reference format:
Huiqiang Chen, Tianqing Zhu, Tao Zhang, Wanlei Zhou, and Philip S. Yu. 2023. Privacy and Fairness in Federated Learning: On the Perspective of Tradeoff. *ACM Comput. Surv.* 56, 2, Article 39 (September 2023), 37 pages.
https://doi.org/10.1145/3606017

1 INTRODUCTION

Machine learning has changed our lives and will undoubtedly bring us more excitement. However, its success is closely tied to the availability of large-scale training data, and as new learning models keep emerging, the demand for more data persists relentlessly. One worrisome issue with collecting massive amounts of data is the risk that presents to privacy. FL [128] has emerged as an attractive learning paradigm to meet privacy requirements.

This article is supported by the Australian Research Council Discovery DP200100946 and DP230100246, and NSF under grants III-1763325, III-1909323, III-2106758, and SaTC-1930941.
Authors' addresses: H. Chen, T. Zhu (corresponding author), and T. Zhang, University of Technology Sydney, PO Box 123 Broadway, Sydney, NSW, Australia, 2007; emails: {huiqiang.chen, tao.zhang-3}@student.uts.edu.au, tianqing.zhu@uts.edu.au; W. Zhou, City University of Macau, Avenida Padre Tomás Pereira Taipa, Macau, China; email: wlzhou@cityu.edu.mo; P. S. Yu, University of Illinois at Chicago, 851 S. Morgan St., Rm 1138 SEO, Chicago, IL 60607, Chicago; email: psyu@uic.edu.

0360-0300/2023/09-ART39 $15.00
https://doi.org/10.1145/3606017

Unlike traditional centralized machine learning, FL trains models in a distributed and parallel way such that different clients collectively train a model with their training data. This technique offers two enormous benefits. First, it saves companies the costly process of collecting large-scale data, because the clients provide their local data. Second, it preserves the client's privacy by keeping data locally. With such benefits, it is no surprise that the industry has already leaped to put FL into practice, such as Gboard [71].

1.1 Privacy and Fairness in FL

Great achievements have been made with FL. However, the paradigm of FL still suffers from ethical issues surrounding data use. One of those ethical issues is privacy. Although raw data never leave the device, the uploaded gradients/parameters still carry local information. Therefore, a trained model could hold the client's data distribution. Consequently, an adversary can infer information about what data are included in the training set [74, 217] or, even worse, reconstruct the training data [58]. As such, the research community is endeavoring to identify all potential privacy risks by launching different privacy attacks [139]. Accordingly, defenses for all these attacks are also being proposed to secure private data [189]. This wargaming between attack and defense is leading us to more private FL environments.

Another ethical issue in FL is fairness, which refers to reducing the model's bias towards disadvantaged groups, such as ethnic minorities, women, or the aged. Fairness in FL is defined at two different levels. The first pertains to *algorithmic fairness* [30, 130], where model output should not skew towards disadvantaged groups defined by some sensitive attributes. The second pertains to *client fairness* [113, 135]. In vanilla FL [128], models trained on the larger dataset are given higher importance during aggregation. Hence, the global model will be optimized to capture the data distributions of clients with a larger dataset. Therefore, the model performance will vary significantly among clients, which imposes unfairness at the client level.

Privacy and fairness are two crucial ethical notions, and violating either of them is unacceptable. However, to date, the research community has primarily considered these two issues separately, yet they are inextricably entwined. For example, it is well known that privacy comes at the cost of accuracy. What is surprising is that the cost is not consistent across all groups as expected, where disadvantaged groups often suffer more of an accuracy decrease than the other groups due to data scarcity [9, 104]. In other words, ensuring privacy can exacerbate the inequities between groups. Fairness, in turn, may negatively affect privacy. For example, to ensure a classification model is fair, the server usually needs to know the underlying distribution of the training dataset to eliminate bias existing in either the training data [49, 90, 91] or the model [61, 210, 211]. This means the client will share more data with the server, increasing the privacy risk.

Therefore, in addition to reviewing privacy and fairness issues in FL, another motivation of this survey is to explore the possible interactions between privacy and fairness and to discuss the relationships between fairness and privacy. It is worth noting that the issue of client fairness adds an extra layer of complexity to the federated setting. To the best of our knowledge, this survey is the first attempt to examine the relationships between privacy and fairness in the federated setting. The comparison between our work and existing works is listed in Table 1.

1.2 Main Contribution

This survey provides a broad view of privacy and fairness issues in FL. We first illustrated that FL is not as private as it claimed to be. Adversarial clients and server have several new attack vectors at their disposal, and we outlined several techniques for preserving privacy against these attacks. Turning to fairness, we explained the two lines of fairness notions adopted in FL and the corresponding debiasing strategies. Last, we discussed interactions between privacy and fairness. Our contributions are as follows:

Table 1. Comparison to Related Surveys on Privacy or Fairness in FL

Reference	Privacy-preserving		Fairness-aware		Interactions between privacy and fairness
	Privacy attack	Defense	Algorithmic fairness	Client fairness	
[199]	✓	✓			
[207]	✓	✓			
[123]	✓	✓			
[88]	✓	✓	✓	✓	
Our work	✓	✓	✓	✓	✓

- This is the first survey that provides a comprehensive overview of privacy, fairness, and the interactions between the two.
- We present a detailed survey of privacy attacks and defenses in FL, discuss how these privacy attacks could damage privacy in FL, and highlight the assumptions and methods of these attack strategies.
- Following a rigorous enumeration of the sources of bias in the FL pipeline, we discuss the fairness notions adopted in FL and summarize fairness-aware FL approaches.
- We point out several future research directions toward training private and fair FL models.

2 BACKGROUND KNOWLEDGE

2.1 Definition of FL

The goal of FL is to train a global model in a distributed way. The objective function is formulated as:

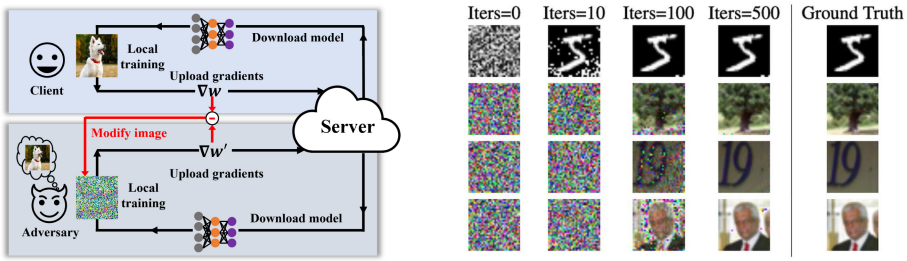
$$\min_{w} f(w) = \sum_{k=1}^{m} p_k F_k(w), \tag{1}$$

where m is the number of clients, $p_k > 0$ is the aggregating weight of client k, satisfying $\sum_k p_k = 1$. $F_k(w)$ is the empirical risk on client k's data. In a trivial setting, p_k is the ratio of local samples to total samples of all clients. The process of FL consists of two stages: the training and inference stages. Three actors are involved: (1) clients, each of which has a local dataset and will use it to contribute to the global model's training by uploading local gradients/parameters, noting that each client's dataset may vary from the others; (2) a server that coordinates the learning process; and (3) users, who will use the final well-trained model.

In each iteration of FL, selected clients download the global model and perform learning algorithms locally. They communicate their updates to the server for aggregation and model updating. This interaction between clients and the server repeats until the model converges. At the inference stage, a well-trained model is deployed to users, where users can infer the model via black-box access. This step is no different from a traditional data center approach.

2.2 Privacy Disclosure in FL

Recent research verified the privacy risk of FL. Adversaries can glean the participants' training data. For example, Zhu et al. [229] fully reconstructed training data from the victim client through the uploaded gradients, as shown in Figure 1(a). In the course of the FL pipeline, several attack interfaces exist for an adversary, who could be the server or a client in FL. The attack could occur during the training or inference stage, within or outside the FL. The attack targets include membership, property, class representative, and the raw data [207].



(a) Overview of reconstruction attack in FL. An adversary iterative optimizes generated sample by matching gradients ∇W.

(b) Attack results on MNIST, CIFAR-100, SVHN, and LFW. The adversary is able to reconstruct the private images of other clients truthfully (reproduced from [231]).

Fig. 1. Reconstruction attack in FL.

2.2.1 Membership Inference Attacks. **Membership Inference Attacks (MIA)** aim to identify whether a given sample was used to train the target model. These types of attacks can pose privacy risks to individuals. For example, confirming a patient's clinical record was used to train a model associated with a particular disease would reveal that patient's health condition. MIA was initially investigated by Shokri et al. [165]. The attack models are essentially binary classifiers. Given an instance $\boldsymbol{X}$ and a target model F_t, the goal of the MIA model is to identify whether or not $\boldsymbol{X}$ is contained within the training dataset D of the target model F_t.

2.2.2 Property Inference Attacks. Property inference attacks aim to recover some property of the training set, which may be irrelevant to the main tasks. Such as the property of "wearing glasses" against a gender classifier or the composition of the training dataset. This kind of attack also leads to privacy issues. With proper prior knowledge, the adversary can infer the presence of a specific sample in the training set.

2.2.3 Model Inversion Attacks. Model inversion attacks aim to recover class-specific features or construct class representatives by accessing the target model and other possible auxiliary information. The recovered data is a representing sample (usually a synthetic sample) that only reflects some aspects of the training data and is not a member of the training set.

2.2.4 Reconstruction Attacks. Reconstruction attacks [38] aim to reconstruct a probabilistic version of samples in the training set. Success in a reconstruction attack is measured by comparing the reconstruction with the original data. If the two are similar, then the attack has been successful. Figure 1(b) [229] shows an example. Unlike the model inversion attacks, the recovered data here is almost the same as the training dataset at the pixel level and belongs to the training dataset.

2.3 Privacy-preserving Techniques

In the realm of FL, plenty of studies have shown how to break the basic privacy assurances, such as determining a client's membership in the training set [139], ascertaining the class representations of the client's training data [74, 170, 188], and, the worst case of all, procuring the raw training data [58]. Several privacy-preserving techniques can help stop these privacy leakages, including cryptographic techniques and the perturbation approach.

2.3.1 Cryptographic Approach. Secure computation is a cryptographic technique in which functions are evaluated based on a set of distributed inputs without revealing additional information, e.g., the parties' inputs or intermediate results. Secure multi-party computation, homomorphic encryption, and secret-sharing are the most common choices for a secure computing platform.

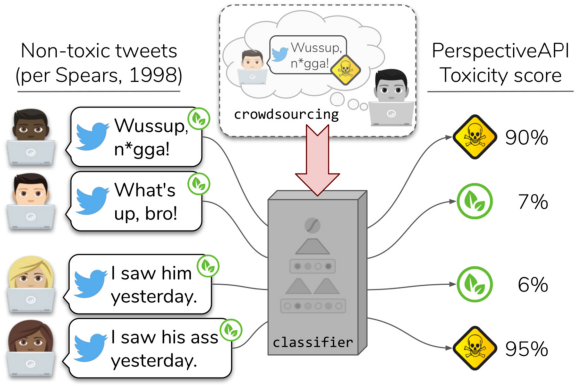


Fig. 2. Racial disparities in classifier predictions on tweets written in African-American English and in Standard American English (reproduced from [156]).

Multi-party computation [203] was first introduced to secure the private inputs of multiple participants while they jointly compute an agreed-upon model or function. Formally, n participants $p_1, p_2, \ldots,$ and p_n can collaboratively compute $y = f(x_1, \ldots, x_n)$, where x_i is a secret input that belongs to participants p_i, This form of secure computing offers both correctness and privacy. After all, no participant learns anything about the others' data other than the final result.

Homomorphic encryption [205] allows certain mathematical operations, such as addition and multiplication, to be performed directly on ciphertexts. These can then be used as the basis for more complex arbitrary functions.

2.3.2 Perturbation Approach. With privacy concerns in mind, one needs to be cautious of how much information about a participating client is revealed during training. The perturbation approach arises as a natural way of preventing information leaks. By injecting the proper amount of artificial noise into the original data, the statistical information calculated from the perturbed data will be statistically indistinguishable from the original data.

There are three types of widely used perturbation techniques: **differential privacy (DP)**, additive perturbation, and multiplicative perturbation. DP, proposed by Dwork [43], is the gold standard. The intuition behind DP is to mask the contribution of any individual user by a sufficient level of uncertainty. A randomized mechanism $\mathcal{M}$ is said to be (ϵ, δ)-differentially private if, for any pair of neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$, and for every set of $S \subseteq Range(\mathcal{M})$, if $\mathcal{M}$ satisfies:

$$\Pr[\mathcal{M}(\mathcal{D}) \in S] \leq \exp(\varepsilon) \cdot \Pr[\mathcal{M}(\mathcal{D}') \in S] + \delta \tag{2}$$

The parameter ϵ is defined as privacy budget, which measures how alike the two adjacent datasets $\mathcal{D}$ and $\mathcal{D}'$ are to each other. A smaller ϵ indicates a stronger privacy guarantee. If $\delta = 0$, then the randomized mechanism $\mathcal{M}$ is degraded into ϵ-DP.

2.4 Fairness in FL

Algorithms are widely used to assist in making recommendations, assessing loan applications, and so on. Several studies have identified the unfairness in different algorithmic scenarios [130]. One example is the hate-speech detector designed to rate the toxicity score of the given phrases to help companies like Twitter recognize harmful speech. The detector relies on a tool called *Perspective*, which is trained on labeled data. The detector behaves differently towards phrases written in African-American English in a racially biased way, as Figure 2 shows [156].

2.4.1 Bias in FL. Fairness can be eroded by bias. According to Olteanu et al. [142], bias can slip into data flows from generation to collection to processing [142]. The distributed learning paradigm of FL brings new and unique challenges to our efforts to build fair models. One challenge is that the **independent and identical distribution (i.i.d.)** assumption no longer holds [88]. In FL, bias can also be introduced by either the client or the server.

- **Client-introduced bias.** Clients can introduce bias in several ways. First, bias exists in a client's local data—prejudice, underestimation, negative legacies, and so on [92]. This bias is then integrated into the global model through client and server interactions. Second, bias can strike when clients are dropped out of the federated setting due to device shutdowns or communication limitations. In these cases, the global model will find it hard to fit the clients' data properly. The massively distributed data also incurs bias. In this case, the client does not have enough data to capture an underlying distribution and, moreover, the underlying distributions of different clients are probably not the same [17, 110, 226].
- **Server-introduced bias.** The server can also add bias. As the coordinator of the learning scheme, the server will sample clients in each round to train a global model with their local data [138]. However, the sampling process is prone to producing bias if it is not done with careful consideration. First, in terms of efficiency, only a fraction of clients are selected in each round [128]. Yet the data distribution of only a few selected clients forms an inadequate representation of the actual population distribution. Second, the sampling may be skewed toward certain clients. For instance, to speed up convergence, the server prefers clients that meet specific criteria [28, 141].

2.4.2 Fairness Notions in FL. To date, researchers have proposed several definitions of fairness; see, e.g., References [130, 181]. These definitions vary from scenario-to-scenario, and it is unlikely that there will ever be one particular definition of fairness that fits all circumstances. Table 2 lists common algorithmic fairness notions adopted in centralized machine learning. At a high level, two families of definitions exist the *individual* notion and *statistical* notion [30].

- **Individual notion.** The individual notions ensure fairness between specific pairs of individuals: *"Give similar predictions to similar individuals."* [42]. Formally, for a set of samples V, a distance metric is defined as $d : V \times V \rightarrow R$ to measure the similarity. A function $\mathcal{M} : V \rightarrow \Delta A$ maps the samples V to the probability distributions over outcomes, and another distance D metric measures the distance between the distributions of outputs. Fairness is achieved if and only if $D(\mathcal{M}(x), \mathcal{M}(y)) \leq d(x, y)$. This family of definitions provides a meaningful guarantee. However, they are at the cost of making significant assumptions, some of which are non-trivial problems in fairness.
- **Statistical notion.** The statistical notions provide fairness assurance at a statistical level. For the protected demographic groups G (such as racial minorities), some statistical measures are required to be equal across all of these groups. These statistical measures include positive classification rates [19, 42, 93], false positive and false negative rates [72, 102], and positive predictive value [29, 210]. Detailed enumeration can be found in References [12, 181]. This family of definitions requires no assumption over data and can be easily verified. However, statistical notions are insufficient as a fairness constraint, which does not give meaningful guarantees to individuals or structured subgroups of the protected demographic groups. Jiang et al. [84] generalized the demographic parity [105] to continuous sensitive attribute.

Apart from algorithmic fairness, which is measured on sensitive attributes, fairness in FL can also be made from a client's view, since clients are naturally grouped by attributes such as geographic location, gender, and income [39]. At a client level, fairness can be evaluated by different metrics.

Table 2. Definitions of Algorithmic Fairness Notions

Fairness notion	Definition	Explanation
Individual Fairness [42]	$D(M(x), M(y)) \leq d(x, y)$	Similar samples receive similar treatment
Eqal Opportunity [72]	$\Pr[\hat{Y} = 1 \mid A = 0, Y = 1] = \Pr[\hat{Y} = 1 \mid A = 1, Y = 1]$	Equal true positive rates for protected/unprotected groups
Equal Accuracy [12]	$\Pr[\hat{Y} = Y \mid A = 0] = \Pr[\hat{Y} = Y \mid A = 1]$	Equal prediction accuracy for protected/unprotected groups
Equalized Odds [72]	$\Pr[\hat{Y} = 1 \mid A = 1, Y = y] = \Pr[\hat{Y} = 1 \mid A = 0, Y = y]$, y $\in \{0, 1\}$	Equal positive rates for protected/unprotected groups
Treatment Equality [12]	Equal false negatives and false positives for protected/unprotected groups	
Demographic Parity [105]	$\Pr[\hat{Y} \mid A = 0] = \Pr[\hat{Y} \mid A = 1]$	Outcome is independent of the protected attribute

Definition 1 (Good-intent Fairness [135]). The training procedure does not overfit a model to any device at the expense of other clients in FL.

This metric improves the worst-case performance. Li et al. [113] took a further step. They tried to ensure a fair FL model for all clients by producing a more uniform model performance across all clients. Fairness is defined as the uniformity of the accuracy distribution across clients in FL.

Definition 2 (Accuracy Parity [113]). Consider two trained models, $f(w)$ and $f(\tilde{w})$. The model that provides the most uniform performance across all clients will also provide the fairest solution to the FL objective in Equation (1).

2.5 Interactions between Privacy and Fairness

Both fairness and privacy are important ethical notions in machine learning and have been extensively studied. However, the majority of current studies in the research community consider fairness and privacy separately. However, the interactions between privacy and fairness are bilateral.

- **Privacy degrades fairness.** Several works have observed inconsistent reductions in accuracy caused by private mechanisms on classification [48] and generative tasks [56]. It turns out that privacy mechanisms affect the underrepresented group more than other groups.
- **Fairness increases privacy risk.** Fairness comes at the cost of privacy. To ensure fairness, a model is trained to perform equally on data from different groups, even though the underrepresented group did not have enough data in the training set, which incurs overfit and increases the privacy risk [21].

3 PRIVACY IN FL

With the advent of FL, many claim that user data are now secure. However, even sharing a small fraction of gradients [5, 164] with a server would raise privacy concerns. In FL, there are several unexplored types of privacy attacks: *membership inference attacks*, *property inference attacks*, *model inversion attacks*, and *reconstruction attacks*. This section will outline these attacks before moving on to mitigation techniques and discussions.

3.1 Membership Inference Attacks

The adversary's goal in FL is to determine whether a given sample belongs to a single client's private training data or of any participants [198]. MIAs occur in different ways in FL.

3.1.1 White-box and Black-box MIAs. Based on the access granted to the adversary, MIAs can be divided into black-box and white-box [139]. In the black-box setting, the adversary can only obtain a prediction vector computed by the target model while the internal parameters remain secret. MIAs in this setting exploit the statistical differences between a model's predictions on its training set versus unseen data [165]. Truex et al. [178] described a systematic approach to constructing a black-box MIA model and the general formulation of each component in the attack model.

However, since the global model is shared with all participants for local training, it is often assumed that an adversary has white-box access in FL. The white-box access renders much more information to the adversary, such as the internal parameters of each layer. This enables the adversary to calculate the outputs of each layer. Nasr et al. [139] designed a deep learning attack model that separately processes the gradients extracted from different layers of the target model and combines this information to compute the membership probability of a target data point.

3.1.2 Training and Inference MIAs. MIAs can be launched during the training stage or once the model is complete in the inference stage in FL.

In the training stage, the adversary could be the server or any client participating in the training. Both characters have white-box access to the global model and can easily save the snapshots of the global model at each iteration during training. In this way, the adversary obtains multiple versions of the target model over time and acquires the updated information to infer private data [149]. In addition to passively collecting the updated information, the adversary may further modify the information to lure the victim clients into revealing more information.

In the inference phase, the FL model is well-trained and fixed. The adversary can only perform an inference attack passively. In this case, MIA in FL resembles that in a centralized setting. The attack's success largely depends on the information that is revealed to the adversary. Melis et al. [131] investigated privacy leaks concerning membership during the inference phase and showed that positions of words in a batch could be revealed from a deep learning model.

3.1.3 Active and Passive MIAs. The adversary can conduct MIAs against the FL model actively or passively. For instance, the server can either adaptively modify the aggregate parameters or honestly calculate the global model and passively conduct MIAs [218].

Melis et al. [131] designed MIAs against models operating on non-numerical data (e.g., natural-language text). An embedding layer is equipped for the target model, transforming the inputs into a lower-dimensional vector representation. The adversary passively saves a snapshot of the joint model parameters w_t. The difference between the consecutive snapshots $\Delta w_t = w_t - w_{t-1} = \Sigma_k \Delta w_t^k$ reveals the aggregated updates from all participants and hence reveals the membership.

Nasr et al. [139] performed active MIAs on FL models by reversing the stochastic gradient descent algorithm and extracting membership information. If the target data point belongs to the training dataset, then the attacker's modifications will be nullified, since the target model will descend the model's gradient for training samples. However, if the target data sample is not used during the training, then the target model will not respond to the attacker's modification. Thus, membership can be deduced. In Reference [74], a malicious client actively mislabels the training sample to fool the victim into releasing private information.

3.1.4 Insider and Outsider MIAs. FL involves two types of actors who can access model information: internal actors (participating clients and the server) and external actors (model consumers and eavesdroppers). Therefore, FL systems must withstand potential adversaries within and outside the protocol.

The inner adversary could be a client or a server. Clients are picked at random to participate in a training round. When training with hundreds or millions of clients, malicious clients are highly

likely involved, who will attempt to deduce the sensitive information of others [74]. The real-time nature of FL added to the inner attacker's strength. For example, Zhang et al. [218] trained a GAN as a malicious client during training to infer the data of other clients in FL. A malicious central server poses a greater threat than a malicious client. Because it can manipulate the global model supplied to victims and obtain more information. Nasr et al. [139] launched MIAs from both the client and server sides and witnessed a higher inference accuracy as a curious central server than as a malicious client.

In addition to internal threats, FL also faces potential attacks from adversaries outside the system. Once the FL training is finished and the model is deployed to users, these users may conduct both black- and white-box attacks, depending on their access.

3.1.5 Discussion. The attacks mentioned above demonstrate the vulnerability of FL to privacy attacks, and these privacy risks stem from two assumptions made within the FL protocols: (1) *The server is trustworthy.* FL gives the server access to each participant's updates in the form of gradients or model parameters containing clients' private information. The server can even purposefully send a modified model to steal information. (2) *Clients are honest.* A malicious client can collect several copies of the global model from the rounds it participates in. In this way, inference phase attacks on data privacy are also plausible during the learning phase. Additionally, adversarial clients may influence and shift the bounds of the model during development rather than just abusing the boundaries of a model's service while it is in production.

3.2 Property Inference Attacks

The adversary in property inference attacks attempts to infer the specific property of the subset of the training dataset. The target property may be irrelevant to the classification task (e.g., "wearing glasses" in a gender classification task) and do not characterize the whole class. The attack is made at the population level as opposed to a single sample in MIA. In terms of when the attack is launched, property inference attacks can be classified as *static* or *dynamic* attacks. The static attack is applied after the training phase has concluded and the target training set is fixed. The dynamic attack typically occurs during the training phase in FL. In this instance, the training set is changing dynamically.

3.2.1 Static Attacks. The research of property inference attacks dates back to Reference [7]. Ateniese et al. performed a property inference attack against Hidden Markov Models and Support Vector Machine based on a meta-classifier. A set of shadow classifiers was trained on a dataset similar to the target model except for the target property. The meta-classifier is trained with shadow classifiers as the input to find the classifiers trained on the dataset with the target property. Ganju et al. [57] extended this attack to a fully connected neural network case. They shared a similar idea, using the gradient of shadow classifiers to train a meta-classifier. Different from Reference [7], their research focuses on improving the attack efficiency by taking permutation invariance into account.

3.2.2 Dynamic Attacks. In every communication round of FL, clients are selected at random. This means the training data is dynamically changing, which weakens the property inference attack because the target property appears unpredictable, thereby diminishing the distinguishability of model updates [187]. Wang et al. [184] explored property inference attacks within the FL framework. Inspired by the relationship between the changing of neuron weights in the output layer and the sample label, the authors proposed three attacks as an eavesdropper to infer the labels' quantity composition proportion. Recently, Wang et al. [187] presented a poisoning-assisted property inference attack in FL from the client's viewpoint, aiming at inferring if and when a sensitive property

emerges. The authors built their attacks around the realization that regular model updates reflect the shift in data distribution and, in particular. A binary classifier is trained to make predictions based on these periodic model updates. A property-specific poisoning attack is proposed to distort the decision boundary of the shared model on target attribute data. Thus, model updates have a better discerning ability to infer target property.

3.2.3 Discussion. The MIAs and reconstruction attacks represent two ends of a spectrum of privacy invasion. Property inference attacks locate in the middle and seek to determine if the attackers' target property is present in the training samples. This type of attack is more complex than MIA, since the target property does not always match the attributes that characterize the classes of the FL model. Nonetheless, a property inference attack poses a greater threat than MIA. Using the real-time nature of FL, the adversary can even infer when the target property appears.

3.3 Model Inversion Attacks

Fredrikson et al. [54] initiated model inversion attacks on tabular data. A subsequent work [53] extended it to the image data. The attack is formulated as an optimization problem to synthesize the input for a given label: $y : \max_x \log T_y(x)$, where $T_y(x)$ is the probability of the model T outputs label y for input x. The access could be black-box [53] or white-box [26, 223].

3.3.1 Black-box Attacks. In the black-box setting, the attacker can only make prediction queries to the model. Fredrikson et al. [53] built attack algorithms following the maximum *a posteriori* principle. Their attack recovered a recognizable image of a person given only API access to a facial recognition system and a specific name of a target person. Yang et al. [202] engineered an inversion model to perform the inversion attacks. The adversary composed an auxiliary set assumed generic enough to retain meaningful information to regularize the ill-posed inversion problem [153]. However, the target model is usually assumed to be simple networks, and the generalization to complex models is not trivial. The inversion problem of a neural network is non-convex, and the optimization suffers minimal local problems, which leads to poor attack performance.

3.3.2 White-box Attacks. In the white-box setting, the attacker has complete knowledge of the model. Zhang et al. [223] sketched a generative model to learn an informative prior from the public dataset. This prior is then used to regulate the inversion problem. Benefiting from this, the authors revealed private training data of DNNs with high fidelity. Chen et al. [26] boosted Reference [223]'s methods. They leveraged the target model to label a public dataset, and a GAN model was trained to distinguish not only real and synthesized samples but also labels. They also modeled the private data distribution to reconstruct representative data points better. The success of model inversion attack benefits from an informative prior.

3.3.3 Discussion. MIAs can be performed with either black-box access or white-box access. When given black-box access, the attack's success heavily relies on the auxiliary dataset, which is assumed to share the same generic features as the private target dataset. Furthermore, the target models in this category are usually simple due to limited access. In the white-box case, the target models extend to DNNs. Most attacks implement GAN to synthesize samples to mimic the private samples regarding the soft labels. This kind of attack is less common compared with reconstruction attacks in FL.

3.4 Reconstruction Attacks

Unlike MIAs, reconstruction attacks attempt to retrieve training data and pose a much more severe threat to privacy. As demonstrated by Aono et al. [5], the gradient of the weights is proportional to that of the bias in the first layer of the model, and their ratio approximates the training input.

Table 3. Comparison between Two Reconstruction Attack Categories

	Theoretical guarantee	Convergence	Running time	Recovered image	Applicability	Insight
Opt-based	No	Local optimal	Slow	With artifacts	No limitation	No
Closed-form	Yes	/	Fast	Original	Limited	Yes

Geiping et al. [58] demonstrated that it is possible to faithfully reconstruct images at high resolution given knowledge of the parameter gradients. Such a privacy break is possible even for deep neural networks. Huang et al. [77] evaluated existing reconstruction attacks and defenses. Gupta et al. [67] extended this attack to text data and successfully reconstructed single sentences with high fidelity for large batch sizes. To date, we know of two kinds of reconstruction attacks, namely, ***optimization-based attacks* (Opt-based)** and *closed-form attacks*. Their differences are detailed in Table 3.

3.4.1 Optimization-based Attack. Raw data can be reconstructed from gradients by solving an optimization problem. Given a machine learning model $f(w)$ and the gradient $g = \frac{1}{b}\sum_{j=1}^{b}\nabla_w L_w(x_j^*, y_j^*)$ computed on a private batch $(x^*, y^*) \in \mathbb{R}^{b\times d} \times \mathbb{R}^b$ with bath size b. The adversary tries to reconstruct $x \in \mathbb{R}^{b\times d}$ as an approximation of the true data x^* by solving the following optimization problem:

$$\arg\min_x \mathcal{L}_{grad}(x; w, g) + \alpha\mathcal{R}_{aux}(x). \tag{3}$$

The first part of Equation (3) pushes the recovered gradients towards the true gradients g, hence deducing a better approximation. The regularization term $\mathcal{R}_{aux}(x)$ is used to incorporate the prior knowledge to further improve reconstruction.

Zhu and Han [229] proposed *DLG* attack with l_2 distance as the reconstruction loss $\mathcal{L}_{grad}$. *DLG* starts with randomly generated dummy samples and then iteratively optimizes the dummy samples and labels until they converge. Finally, *DLG* achieves pixel-wise recovery accuracy for image classification and token-wise recovery accuracy for a masked language model. Zhao et al. [224] improved Zhu and Han's work [229] on convergence speed and reconstruction fidelity by leveraging the relationship between the ground-truth labels and the signs of the gradients.

The optimization problem Equation (3) is often under-determined [81], as the information in the gradients g is usually insufficient to recover the training data x^*, even when the gradient size is substantially bigger than the input data dimension. As demonstrated by Zhu and Blaschko [228], when the learning model is huge, there may be a pair of separate data sharing the same gradient.

In response to this, one may introduce prior knowledge as a regularization term to narrow the search space, making it more consistent with the underlying distribution of training data. Yin et al. [206] utilized the local batch norm as the regularization term, since adjacent pixels in natural photographs are likely to have comparable values. They achieved precise recovery of the high-resolution images on complex datasets, deep networks, and large batch sizes. Hatamizadeh et al. [73] extended Reference [206]'s approach to vision transformers and discovered that, because of the attention mechanism, vision transformers are substantially more sensitive than previously researched CNNs. In the image domain, total variance is another choice [58, 188].

Selecting a proper loss function can also contribute to attack efficiency. By combining mean square error and Wasserstein distance [6], Ren et al. [152] achieved a better reconstruction result than References [224, 229] in terms of batch size and reconstruction fidelity. Geiping et al. [58] adopted cosine distances to better capture the observation that the angle between two data points quantifies the change in prediction. With this method, they rebuilt a single high-resolution image and a series of low-resolution photos with a maximum batch size of 100. Jeon et al. [81]

systematically investigated ways to best utilize gradients, including using them to extract prior information. Wei et al. [191] conducted a thorough analysis of how different hyper-parameter setups and settings for attack algorithms influence the effectiveness of the attack and its cost. In an effort to investigate the worst-case attack and evaluate the effectiveness of defense methods for reconstruction attacks, Balunovic et al. [10] formulated the gradient leakage problem in a Bayesian framework and analyzed the condition for a Bayes optimal adversary.

3.4.2 Closed-form Attack. In an attempt to provide a theoretical understanding of how and when gradients lead to the remarkable recovery of original data, several studies investigated the possibility of recovering the input of a learnable affine function from gradients [5, 47, 151]. Aono et al. [5] initiated the closed-form attack based on gradients. In certain circumstances, an honest-but-curious server could directly calculate individual data from the gradients uploaded by clients [5, 228].

Consider a fully connected layer $Wx+b=z$ with $l=l(f(x),y)$ as the loss function, where x and z are the input and output vectors, respectively. Private data x can be derived from l's gradients w.r.t. W and b, i.e. : $x^T = \frac{\partial l}{\partial W} \oslash \frac{\partial l}{\partial b}$. Here, $\oslash$ denotes entry-wise division. The assumption of a fully connected layer holds for the last prediction layers in many popular architectures. As a result, the prediction modules' input, which is the output of the preceding layers, can be rebuilt. These outputs typically include some information about the training data, making them vulnerable to attackers. In this light, the ability to recover ground truth label information from the gradients of the final fully connected layer, as stated in Zhao et al. [224], is very intriguing.

Despite its inspiring nature, Aono et al.'s work [5] has some limitations. First, it does not apply to convolutional neural networks due to a mismatch in dimensions. Second, it cannot deal with batch inputs. For the batch input $\{x_j\}_{j=1}^b$, all derivatives are summed over the batch dimension b and the recovered $\bar{x}$ is merely proportional to the average of batch inputs $\sum_{j=1}^b x_j$.

To fix the dimension mismatch issue, Zhu and Blaschko [228] converted the convolutional layer into a fully connected layer using circulant matrix representation [62] of the convolutional kernel [47]. The gradients of each layer were interpreted as the *gradient constraints*. Finally, they recursively reconstructed the layer-wise input. However, their implementation can only recover low-resolution images in settings where the batch size equals 1. For the batch input, the algorithm returns a linear combination of the training data.

To address these difficulties with the batch size, Fowl et al. [52] suggested making minor but malicious changes to the global model to reconstruct the client's data from a batch of gradient updates. The key idea is to separate the batch data by some quantity h, such as image brightness. To this end, an imprint module is added to the global model, which acts as a filter that separates a batch of samples based on quantity h. Qian and Hansen [151] found bias term in the output layer is the key to the success of reconstruction. A fully connected neural network requires just one node in one hidden layer for single-input reconstruction. In contrast, mini-batch reconstruction requires that the hidden units exceed the input size. Pan et al. [144] conducted an analytic investigation of the security boundary of the reconstruction attacks. Given a batch input, the secure/insecure boundary of the reconstruction attack was characterized by the number of **Exclusively Activated Neurons (ExANs)**, where the more ExANs, the more likely the attack's success.

3.4.3 Discussion. The closed-form attacks outperform optimization-based attacks in several aspects. First, the closed-form attacks provide a theoretical guarantee of convergence. In contrast, optimization-based attacks suffer from the local optimum problem, since a non-convex optimization may not always converge to a correct solution [58]. Further, optimization-based attacks are sensitive to initialization [229], whereas closed-form attacks are not. Second, the

Table 4. Comparison of Reconstruction Attacks in FL

Method	Objective function		Maximal batch size	Opt-based/ Closed-form	Theoretical guarantee	Additional information
	$\mathcal{L}_{grad}$	$\mathcal{R}_{aux}$				
iDLG [224]	l_2 distance	/	8	Opt-based	No	No
DLG [229]	l_2 distance	/	8	Opt-based	Yes	No
Inverting gradients [58]	Cosine similarity	Total variance	100	Opt-based	Yes	Local updates; BN statistics
[191]	l_2 distance	Label-based regularizer	8	Opt-based	Yes	No
SAPAG [186]	Gaussian kernel based function	/	8	Opt-based	No	No
R-GAP [228]	Recursive gradients	/	5	Closed-form	No	No
Theory-oriented [144]	l_2 distance	l_1 distance of feature map	32	Closed-form	Yes	Exclusive activated neurons
GradInversion [206]	l_2 distance	Group consistency	48	Opt-based	No	BN statistcs
CAFÉ [86]	l_2 distance	Total variance	100	Opt-based	Yes	Batch indices
GIAS&GIM [81]	Negative cosine	l_2 distance in latent space	4	Opt-based	No	No
Imprint module [52]	One-shot mechanism	/	16384	Closed-form	Yes	CDF
GradViT [73]	l_2 distance	Image prior; Auxiliary Regularization	64	Opt-based	No	Auxiliary networks

deterministic algorithms run by closed-form attacks are faster than optimization-based attacks. Third, closed-form attacks recover the data more accurately, while optimization-based methods, like GradInversion [206], recover data with artifacts. Jin et al. [86] made a good comparison between different reconstruction attacks in FL. Table 4 summarizes their finding and includes some additional results from this study.

3.5 Privacy-preserving Techniques

Privacy-preserving machine learning approaches can be roughly classified as cryptographic approaches and perturbation approaches. Cryptographic approaches enable computation over encrypted data and provide rigorous privacy guarantees in the training process. However, they come at a high computational cost compared to the non-encryption alternatives [197]. This computation overhead limits their application in some learning scenarios, particularly in deep neural networks with huge amounts of parameters. As a result, most state-of-the-art privacy-preserving methods are perturbation-based. The perturbation can be accomplished by adding artifact noise into the dataset, such as DP mechanism [59, 189]; or by representing the raw dataset with a surrogate dataset [158, 192] or abstracting the dataset via sketch techniques [68, 111].

3.5.1 Cryptographic Approaches. Secure multi-party computation is a sub-field of cryptography that executes calculations on data dispersed among multiple parties in such a way that the computation results are only revealed to the participants [203]. It can take the form of **homomorphic encryption (HE)** or secret-sharing.

As one of the de facto privacy-preserving solutions, HE provides perfect privacy protection in the face of a malicious server. It allows clients to encrypt their updates in such a way that the server may directly aggregate ciphertexts without divulging anything about the plain text

underneath. The downside is that encryption followed by decryption will inevitably impose both a computation and a communications overhead. Phong et al. [148] used *additively homomorphic encryption* to ensure no information was leaked to a malicious server. The encrypted aggregation was formulated as follows:

$$\mathbf{E}(\mathbf{W}_{\mathbf{global}}) := \mathbf{E}(\mathbf{W}_{\mathbf{global}}) + \mathbf{E}(-\alpha \cdot \mathbf{G}_{\mathbf{local}}), \tag{4}$$

where $\mathbf{E}$ is a homomorphic encryption operator that supports addition over ciphertexts, and $\mathbf{G}_{\mathbf{local}}$ is the aggregated gradient. The decryption key is public to the clients and private to the server, and thus, the client's information is secured. Due to the additively homomorphic property of $\mathbf{E}$, each client is still able to receive the correct updated model W_{global} via decryption.

$$\mathbf{E}(\mathbf{W}_{\mathbf{global}}) + \mathbf{E}(-\alpha \cdot \mathbf{G}_{\mathbf{local}}) = \mathbf{E}(\mathbf{W}_{\mathbf{global}} - \alpha \cdot \mathbf{G}_{\mathbf{local}}) \tag{5}$$

However, the amount of data transferred between clients and the server is inflated by two orders of magnitude over the vanilla setting [148]. To reduce the communication load, Zhang et al. [215] chose a **distributed selective stochastic gradient descent (DSSGD)** method in the local training phase to achieve distributed encryption and reduce the computation costs. Zhang et al. [213] presented a BatchCrypt as a simple batch encryption technique. Clients first quantize their local gradients and then encode a batch of quantized updates into a long integer. As a result, the communication overhead is reduced by up to 101 times. Jiang et al. [85] further reduced communication overhead by sending only a sparse subset of local states to the server.

Another drawback is that all participants share the same private key for decryption, since homomorphic operations require all values to be encrypted with the same public key, which degrades privacy protection in the face of malicious clients. To counter this problem, Park and Lim [147] sketched a privacy-preserving FL scheme based on a distributed homomorphic cryptosystem that allows clients to have their own unique private key for the homomorphic encryption scheme.

Secret sharing [159] is another kind of cryptographic technique. It splits a secret data $\mathcal{D}$ into n pieces such that the secret $\mathcal{D}$ can be easily reconstructed with at least k pieces. However, any set containing less than k piece reveals no information about $\mathcal{D}$. It enables the server to aggregate at least a certain number of clients' updates without disclosing any individual client's contribution. Bonawitz et al. [14, 136] proposed a secure aggregation method for FL based on t-out-of-n secret-sharing. The key idea is to mask the raw data in a symmetric way. Thus, when aggregated by the server, the introduced noise will be nullified. Liu et al. [118] incorporated secure sharing into their federated transfer learning framework to protect privacy. Based on an investigation of how secure aggregation parameters influence communication efficiency, Bonawitz et al. [15] used quantization to build a communication-efficient secure aggregation scheme. So et al. [168] designed Turbo-Aggregate, which leverages additive secret-sharing and Lagrange coding to reduce the secure aggregation overhead. Shao et al. [160] shared a similar ideal, utilizing Lagrange coding to secretly share private datasets among clients.

Even though FL based on a homomorphic encryption scheme can prevent privacy leaks during training, it remains vulnerable to attacks in the inference stage. The trained model embodies the distribution of training data to a certain extent, and the privacy risk still exists. Model inversion attack gives such an example. Given the white-box access to a trained model, Zhang et al. [223] successfully discovered the sensitive features x associated with a specific label y.

3.5.2 Perturbation Methods. Due to its theoretical guarantee of privacy protection and its low computational and communication complexity, the DP technique [41] has emerged as the most popular choice for privacy protection among a variety of options. In differential privacy, a proper amount of noise is added to the raw data [66], the model [59, 117], the output [22], or the gradients [171] to protect privacy.

Geyer et al. [59] applied a differentially private mechanism to the FL scenario, where they approximated the averaging operations with a randomized mechanism that provided client-level privacy. Truex et al. [177] presented an alternative approach that draws on both DP and secure multi-party computation. The clients and the server communicate through a secure channel. Upon receiving a request from the server, the clients upload their answers following the principles of DP. Xu et al. [194] took a different approach. They approximated the objective function of a regression problem via polynomial representation and then added Laplace noise to the polynomial coefficients to protect privacy. Khalili et al. [97] exploited an exponential mechanism [129] to privately select applicants based on the qualification scores predicted by a pre-trained model.

One concern with perturbation techniques is the tradeoff between privacy, accuracy, and convergence. A significant noise perfectly protects privacy at the cost of accuracy and convergence. Conversely, a weak noise is futile to privacy attacks [74]. Wei et al. [189] conducted a theoretical analysis of the convergence behavior of FL with DP and identified the tradeoff between convergence performance and privacy protection levels.

Many studies have been published on ways to deal with this tradeoff [47, 164, 221]. Shokri and Shmatikov [164] suggested randomly selecting and sharing a small fraction of gradient elements (those with large magnitudes) to reduce privacy loss. Fan et al. [47] leveraged element-wise adaptive gradient perturbations to defeat reconstruction attacks and maintain high model accuracy. In a similar manner, Wei and Liu [190] used dynamic privacy parameters, introducing noise with a greater variance at the beginning of training and progressively decreasing the amount of noise and variance as training progresses.

Huang et al. [78] proposed *InstaHide*, a combination of cryptographic and perturbation approaches to provide rigorous privacy protection at the cost of minor effects on accuracy. *InstaHide* encrypts the raw image by mixing it with multiple random images from a large public dataset. After that, it randomly flips the signs of the pixels before using it to train the model.

Yang et al. created [200] NISS to avoid the tradeoff between accuracy and privacy by permitting clients to collaborate on reducing the total amount of injected noise. In particular, each client's noise is neutralized and distributed to other clients. Theoretically, if all clients are trustworthy, then the locally introduced noise can be perfectly offset by the server's aggregation, completely avoiding the privacy accuracy tradeoff. A similar idea can be found in Yang et al. [201].

3.5.3 Trusted Execution Environment. Some researchers use **Trusted Execution Environments (TEEs)** such as Intel SGX and ARM TrustZone to secure ML training in untrusted environments [64, 133, 174]. With hardware and software safeguards, TEEs secure critical code from other programs. Compared with purely cryptography methods, TEEs provide much better performance, since it only requires extra operations to create the trusted environment and communicate between trusted and untrusted components. Gu et al. [64] partitioned DNN models and solely encased the first layers in an SGX-powered TEE to protect input information. Hynes et al. [79] investigated speeding up the training using **Graphics Processing Units (GPU)**. Tramer et al. [174] shared the same concept and offered effective privacy-preserving neural network inference utilizing trusted hardware that delegated matrix multiplication to an untrusted GPU. However, this work does not translate well to FL due to the possible adversary server and limited computation power of the client device. To remedy this, Mo et al. [133] advocated using the TEE of client devices in tandem with model partitioning to defend against MIA. The model is divided into two halves, and the final layers are calculated within TEE. Kato et al. [96] proposed to combine DP with TEE in FL in the presence of an untrusted server. The models are aggregated within the TEE of the server's device [27].

3.5.4 Discussion. Table 5 summarized and compared the existing defense techniques. Cryptographic approaches preserve privacy to a great extent while suffering from computational

Table 5. Privacy-preserving Methods in FL

Attack	Defense method	Rationale	Advantage	Disadvantage
RA	HE [5, 85, 162]	Gradients are encrypted	Accurate	1. Vulnerable if there are multiple colluding entities; 2. Ineffective at inference
	Secret sharing [14, 136, 160]	Hiding information about clients' individual update, except for their sum	1. Accurate; 2. Robust to users dropping out	Ineffective at inference
	Variational bottleneck [158]	Using surrogate gradient to protect privacy	Keep training process and performance intact	Limited to optimization-based attack
	Gradient compression [114, 172, 229]	Compressing gradients to prevent reconstruct private data by matching gradients	1. Easy to implement; 2. Reduce communication	Requires considerable noise, degrades model performance, and increases convergence time
RA, MIA, and PIA	DP [25, 124, 191]	Hiding private information by injecting noise to the raw data, model, or output	1. Easy to implement; 2. Long-term protection	
	TEEs [96, 133]	Isolating part of networks from the untrusted environments	Reduce computation	Limited memory space

RA: Reconstruction attack; MIA: Membership inference attack; PIA: Property inference attack.

complexity and are less feasible. The perturbation approaches trade off privacy for model performance. Several inspiring works demonstrate that it may be possible to avoid that tradeoff through either client collaborations to neutralize locally added noise on the server side or by using a surrogate dataset to protect the raw data without adding noise. The cryptographic approaches only ensure that no information will leak during training. They do not protect privacy during the inference stage. In contrast, the perturbation approaches (e.g., DP) protect privacy in both the training and inference stages. One may combine cryptographic and perturbation approaches to obtain better privacy protection throughout the machine learning pipeline.

3.6 Discussion of Privacy Attacks and Defenses in FL

This section reviews existing privacy attacks and defense approaches in FL. Table 6 summarized the existing privacy attacks in FL. From the attacker's perspective, FL differentiates from the centralized counterparts in sever aspects: (1) *The active attacker in FL.* Due to the collaboration between clients and the server, an adversary could actively attack for victim's private data. For example, the attacker may maliciously reverse the gradients [139] or mislabel the training sample [74] to neutralize the benign clients' efforts and fool them into revealing more information about their private data, hence, making them more vulnerable compared to centralized machine learning. (2) *The real-time nature of FL strengthens the attacker's ability.* During the training process, the adversary could adaptively change their strategy to infer the victim's private data. As a result, the adversary can even infer a specific clients' data [188] when the target features appear in FL [184, 187], which is way more severe than centralized setting. (3) *Gradients are shared between clients and server.* Unlike the centralized counterpart, where the adversary could at most access the white-box access to the target model, in FL, gradients are repeatedly shared between clients and the server, which enables gradient-based privacy attacks. As shown by References [206, 229], the malicious server can reconstruct clients' training data at the pixel level by minimizing the distance to the target gradients.

From the defender's perspective, protecting privacy in FL is also different from that in the centralized scenario. (1) *The malicious attacker could be the server or any client.* FL allows clients to keep private data local. A central server is designed to orchestrate the training process, which

Table 6. Comparison of Privacy Attacks in FL

Attack	Ref.	Access	Attack interface	Assumptions	Key technique
Membership Inference Attack	[165]	BB	Confidence vector	Knowledge about population data	Inferring from the discrepancies of predictions on training set versus unseen data
	[139]	WB	Model parameters	A significant portion of the training data	Reversing the SGD algorithm
	[108]	WB	Model parameters	A proxy dataset sampled from the ground-truth distribution	Inferring from parameter differences between the target and proxy model
Property Inference Attack	[7]	WB	Model parameters	1. Knowledge about training data structure; 2. Access to the ground-truth distribution	A meta-classifier to infer properties from multiple shadow classifier parameters
	[187]	WB	Model updates	Adversary can manipulate more than one device in FL	Inferring other clients' data from the periodic model updates
	[184]	WB	Model updates	1. Client's average label count; 2. Number of samples per label	Inferring from the layer neuron weight changes
Model Inversion Attack	[53]	BB/WB	Confidence vectors / Model parameters	1. Side information; 2. Simple networks	Optimizing the input to maximize confidence vectors subject to the classification matches the target
	[202]	BB	Confidence vectors	1. Auxiliary dataset retains meaningful prior information; 2. Simple networks	An inverse model approximates the mapping between predictions and images
	[223]	WB	Feature extractor	An auxiliary dataset retains meaningful prior information	1. Distilling prior knowledge from an auxiliary dataset via GAN; 2. Optimizing the generated image to maximize likelihood
Reconstruction Attack	[188]	WB	Client updates	1. Shallow target model; 2. Low-resolution image	A GAN with multi-task discriminator to enhance fidelity and identify client
	[58]	WB	Client updates	1. Honest-but-curious server in FL; 2. Small batches	Optimizing the image to get a similar change in model prediction as the target images
	[229]	WB	Gradients	1. Small image size; 2. Single batches; 3. Target model is twice differentiable	Jointly optimizing inputs and labels to match gradients
	[73]	WB	Gradients	Auxiliary networks provide image prior	Optimizing inputs to match the target gradients with image prior and auxiliary regularizer
	[206]	WB	BN layers	1. BN layers in target model; 2. No repeating labels in a batch.	Optimizing the input to match the statistics of BN layers of the target model

BB: black-box; WB: white-box.

complicated privacy protection. The adversary could be the server [58, 152, 188] or the client [74]. The malicious adversary is able to infer the target client's privacy passively or actively; for example, sending the modified global model to the target client to probe private data [52]. This brings challenges to defending against potential privacy attacks. DP is a prevailing choice, but it degrades performance. The cryptographic approaches, such as HE and MPC, retain both privacy and performance at the cost of computation overhead, which is a more severe issue in FL, since most clients' devices are limited in computation power. (2) *Training and inference stage privacy attacks.* Different from centralized machine learning, where the major privacy leakage happens at the inference stage, i.e., malicious users probe private training data by inferring the target model. In FL, the attacks could happen during or after the training. This requires the defenders to be aware of both possibilities. The cryptographic approaches provide provable privacy protection

during the training stage; however, they fail at the inference stage, since training distribution is embedded in the trained model's parameters. The perturbation approaches, e.g., DP, provides long-term protection and covers both the training and inference stages. One can hide sensitive information from adversaries by adding appropriate noise to the training data.

4 FAIRNESS IN FL

Fairness, as discussed in the centralized setting, is mainly defined at either the group level [72, 105] or the individual level [42]. In the FL scenario, fairness has a broader definition. Beyond the long-established *algorithmic fairness* [72, 105], *client-level fairness* [113, 126, 135, 208] arises as a new challenge in FL.

4.1 Algorithmic Fairness

Algorithmic fairness is commonly used to describe the discrepancies in algorithm decisions made across distinct groups as defined by a sensitive attribute. FL often involves a deep neural network with redundant parameters and is pruned to overfit the privileged groups. Various debiasing methods have been devised for different applications, including machine learning [11, 13, 18, 210, 219], representation learning [120, 212], and natural language processing [16, 51, 127, 225]. These methods vary in detail but share similar principles. Following the data flow, debiasing methods can be grouped into *pre-processing*, *in-processing*, and *post-processing* categories, which address the discriminate issues at three distinct stages of the data's handling [34].

4.1.1 Pre-processing. Pre-processing tries to remove the underlying discrimination from the data typically by (1) altering the values of the sensitive attributes/class labels; (2) mapping the training data to a new space where the sensitive attributes and class labels are no longer relevant [49, 90, 91]; or (3) reweighting the samples in the training dataset to compensate for skewed treatment [91].

Intuitively, by training a classifier on discrimination-free data, it is likely that the resulting predictions will be discrimination-free. Inspired by this idea, Kamiran and Calders [91] proposed three types of pre-processing solutions to learn a fair classification: *messaging, reweighting, and sampling*. Feldman et al. [49] investigated the problem of identifying and removing disparate impacts in the data. Xu et al. [196] proposed FairGAN, which generates fair data from the original training data and uses the generated data to train the model. Abay et al. [1] proposed two reweighting methods for the FL setting [91], *local reweighting* and *global reweighting with DP*.

Notably, these pre-processing techniques require access to the training data, which violates the privacy principles of FL. As a result, these types of techniques can only be deployed locally on each client. However, in the presence of data heterogeneous among clients, local debiasing cannot provide fair performance for an entire population [32].

4.1.2 In-processing. In-processing modifies traditional learning algorithms to address discrimination [11, 61, 92, 210–212], such as adding a regularization term to the loss function. Berk et al. [11], for example, incorporated a family of fairness regularizers into the objective function for regression problems. These regularizers span the range from notions of group fairness to individual fairness. They also create a tradeoff between accuracy and fairness.

Another in-processing option is imposing constraints. Zhang et al. [212] used a GAN to constrain the bias in a model trained on biased data. During training, the scheme simultaneously tries to maximize the accuracy of the predictor while minimizing the ability of the adversary to predict the protected variable. In FL, Gálvez et al. [55] studied the notion of group fairness as an optimization problem with fairness constraints. Papadaki et al. [145] formulated a min-max optimization problem to investigate group fairness in scenarios where population data were distributed across

clients. Ezzeldin et al. [45] replaced the aggregation protocol FedAvg with FairFed, which adaptively updates the aggregating weights in each round to improve group fairness. Clients whose local measurements match the global fairness measure are given preferential treatment. Khedr et al. [98] add a regularizer term to minimize the average loss in fairness across all training data.

4.1.3 Post-processing. Post-processing addresses discrimination issues after the model is trained and does not need to change the training process. The general methodology of post-processing algorithms is to take a subset of samples and change their predicted labels to meet a group fairness requirement [13, 19, 20, 72, 119, 150].

Hardt et al. [72] proposed a post-processing technique to construct a non-discriminating predictor $\tilde{Y}$ from a learned discriminatory binary predictor $\hat{Y}$. Only access to the prediction $\hat{Y}$, the protected attribute A, and target label Y in the data are required, while details of the mapping of features X to prediction $\hat{Y}$ are not needed. Canetti et al. [20] and Pleiss et al. [150] shared the key characteristics as Hardt et al.'s [72] work. Lohia et al. [122] designed a post-processing method to increase both individual and group fairness. Salvador et al. [154] introduced a conditional calibration method for fair face verification. Their method clusters images into different sets and assigns distinct thresholds to different sets.

4.1.4 Discussion. Three different kinds of debiasing methods are at hand in centralized machine learning. However, solutions in the centralized setting cannot be applied directly in the FL scenario due to limitations with the training data. More specifically, in federated settings, the clients usually have limited amounts of data. Hence, a single client cannot accurately represent the true distribution over all clients. Consequently, debiasing data before training is not an option. Another limitation is that direct access to local data is prohibited on the server side. Nevertheless, canny researchers have found inspiration from and workarounds to these issues. Gálvez et al. [55], for example, bypassed this access restriction by using statistics to guide the model's training instead of the raw data.

4.2 Client Fairness

Client fairness in FL is another different fairness notion than algorithmic notions. Ideally, the models produced from FL should capture clients' data distributions and generalize well when deployed on the client side. However, data distribution usually varies among clients. As a result, the global model has inconsistent performance on different clients' dataset. At the client level, an FL protocol is considered to be fair if the performance fluctuates within a limited range, i.e., the variance in the model's performance across clients falls under a predefined threshold. To this end, two lines of research exist to mitigate fairness issues in FL. These are the *single model approach* and the *personalized models approach*.

4.2.1 Single Model Approach. The single model approach trains a single global model for all clients as a standard FL scheme. Here, the focus is on solving any statistical heterogeneity during the training phase rather than smoothing the distribution difference.

- **Data augmentation** is a straightforward solution to statistical heterogeneity. It increases data diversity on the client side. Several researchers have studied ways to enhance the statistical homogeneity of local data in FL [70, 82, 226]. Zhao et al. [226] suggested a data share scheme, which creates a globally shared dataset that is balanced by class. The experiment shows a 30% improvement on accuracy with only 5% globally shared data. Jeong et al. [82] proposed *FAug*. Clients first collectively train a GAN model, which is then distributed to clients to augment their local data towards yielding an i.i.d. dataset.

- **Client Selection** is another strategy that focuses on sampling data from a homogeneous distribution. Wang et al. [183] proposed a control framework to actively select the best subset of clients in each training round. In Yang et al.'s [198] method, the local data distribution is estimated first by comparing local updated gradients and gradients inferred from a balanced proxy dataset. The client selection algorithm based on a combinatorial multi-armed bandit was designed to minimize the effect of class imbalances.
- **Agnostic approach** trains a robust model against a possible unknown testing distribution. Mohri et al. [135] modeled testing distributions as an unknown mixture of all m clients' data. The global model is optimized for all possible target distributions. This makes the global model more robust to an unknown testing distribution. Du et al. [39] introduced a fairness constraint into Mohri et al.'s method [135] and proposed *AgnosticFair*, a fairness-aware FL framework. Their method can provide both *Good-intent fairness* and *demographic parity*.
- **Reweighting** tries to train a fair model by assigning suitable aggregating weights p_k in Equation (1) to clients. Inspired by α-fairness notions [107, 134], Li et al. [113] sketched ***q-Fair FL* (q-FFL)** to foster fairer accuracy distribution across all clients by up-weighing clients with lower performance during aggregation. Huang et al. [76] shared a similar idea where, for each round of aggregation, clients with lower accuracy or less training participant times are assigned higher aggregation weights.

4.2.2 Personalized Models Approach. Instead of smoothing the statistical heterogeneity, in *personalized FL*, multiple distinct models are trained for clients with different data distributions. A global model is first trained collaboratively and then personalized to clients using private data. In this way, clients can benefit from other clients' data and solve the issue of statistical heterogeneity. Mansour et al. [125] designed and analyzed three approaches to learning personalized models to learn personalized models. Kulkarni et al. [103] conducted a brief overview of personalized FL. Chen et al. [23] provided a comprehensive benchmark of various personalized FL methods. Tan et al. [173] systematically reviewed this topic and classified personalized FL techniques in terms of data-based and model-based approaches. Here, we summarize their conclusions.

- **Multi-task learning** treats building models for each client as different tasks. Smith et al. [167] pioneered this approach and explored personalized FL via a multi-task learning framework. References [3, 115] followed this principle. Dinh et al. [37] proposed FedU, which incorporates a Laplacian regularization term into the optimization problem to leverage relationships between clients.
- **Model interpolation** trains local and global models simultaneously, where the global model is used for its generalization ability, and the local model is used to improve local performance. Hanzely and Richtárik [69] formulated an optimization problem that learns a mixture of the global and local models. The local model is trained solely on each client's private data. Softly enforced similarity from multi-task learning is borrowed to discourage the local model from departing too much from the mean model. Deng et al. [35] and Mansour et al. [125] adopt a similar formulation to determine the optimal interpolation of the local and global models. In Zhang et al.'s [220] work, clients are given access to multiple models uploaded by other clients to evaluate how much they will benefit from these models. An optimal combination is then used as a personal update. Lin et al. [116] investigated the tradeoffs between local and global models.
- **Parameter decoupling** learns local parameters as an independent task performed locally. The local model is designed to assist in personalizing the global model to local distributions. Liang et al. [115] devised the local-global federated averaging algorithm, which jointly learns compact local representations for each client and a global model across all devices. Chen and

Chao [24] decomposed an FL model as a generic predictor, which is trained globally, along with a personalized predictor that is trained locally. The personalized predictor is formulated as a lightweight, adaptive module on top of the generic predictor.

- **Transfer learning** is a practical training paradigm that leverages knowledge from a source domain to help train a model in a target domain. The performance of transfer learning depends on the similarity between the two domains. Federated transfer learning was first introduced by Liu et al. [118]. Since clients in the same federation usually share the same domain, an FL scheme would make a suitable partner for transfer learning. Li and Wang [109] subsequently proposed FedMD, which combines transfer learning and knowledge distillation. Each client performs transfer learning by training a model to converge on a public dataset and subsequently fine-tune it on local data.
- **Clustering** arranges clients into different groups and trains a specific model for each group. Ghosh et al. [60] iteratively determine the membership of each client to a cluster and optimize each of the cluster models via gradient descent in a distributed setting. Sattler et al. [157] cluster clients according to the cosine similarity between the clients' gradient updates. This allows clients with a similar distribution to profit from one another while minimizing detrimental interference from others. In Briggs et al.'s [17] method, a clustering step is periodically inserted into the training process to cluster clients based on their local updates. The clusters are then trained individually and in parallel on specialized models. Mansour et al. [125] proposed hypothesis-based clustering, partitioning clients into q clusters and finding the best hypothesis for each cluster.
- **Regularization** prevents overfitting when training models and has been used in several studies to remedy the weight divergence problem in FL settings. Li et al. [112] introduced a proximal term that considers the differences between global and local models to limit the effect of local updates. Yao et al. [204] considered parameter importance in the regularized local loss function by using elastic weight consolidation [101]. In addition, a regularization term is introduced to penalize the deviation of the local model from the global model.
- **Meta-learning** aims to leverage prior experience with other tasks to facilitate the learning process. The resulting models are highly adaptable to new heterogeneous tasks [50, 140]. Fallah et al. [46] studied a personalized variant of FedAvg based on model-agnostic meta-learning formulation. The proposed Per-FedAvg algorithm looks for an initial model that performs well after one step of the local gradient update on each client's data. Others have interpreted FedAvg as a meta-learning algorithm, breaking it into two stages of training and fine-tuning to optimize personalized performance and model convergence [83, 99].

4.2.3 Discussion. In addition to algorithmic fairness, client fairness is another concern in the FL community. Table 7 enumerated various works on these two topics. Regarding client fairness, the single model approach focuses on smoothing data heterogeneity, where it is easy to implement and can be added to the general FL paradigm, since it only needs modest modification. On the downside, the single model approach is less effective than personalized approaches in terms of capturing local data distribution and may be insufficient when the data distributions vary significantly between clients. Additionally, the single-model approach does not allow clients to customize their models.

4.3 Discussion of Fairness in FL

There are two definitions of fairness in FL: *client fairness* and *algorithmic fairness*. *Algorithmic fairness* has been extensively studied in centralized machine learning. These algorithms presuppose centralized access to data, however, one virtue of FL is data never leaves the device. This means neither the server nor any client gains centralized access to the training data. Therefore, generalizing

Table 7. Summary of Fairness-aware FL

Reference	Single Model	Personalized Model	Algorithmic Fairness	Client Fairness	Method
[70, 82, 226]	✓			✓	Data Augmentation
[40, 183, 198]	✓			✓	Client Selection
[145]	✓		✓		Agnostic approach
[39]	✓		✓	✓	Agnostic approach
[75, 135]	✓			✓	Agnostic approach
[45]	✓		✓		Reweight
[76, 113]	✓			✓	Reweight
[55, 94, 209]	✓		✓		Regularization
[17, 60, 125, 157]		✓		✓	Cluster
[35, 69, 125, 220]		✓		✓	Model interpolation
[3, 37, 115, 167]		✓		✓	Multi-task learning
[24, 115]		✓		✓	Parameter decoupling
[109, 118]		✓		✓	Transfer learning
[112, 163, 204]		✓		✓	Regularization
[46, 83, 99, 166]		✓		✓	Meta-learning

the fair learning algorithms to FL is not trivial. On the one hand, data are stored locally in FL. The server cannot directly access the local data of clients. Hence, server-side debiasing is not a viable solution. On the other hand, debiasing on the client side is ineffective due to the inadequate data, which can hardly represent the global data distribution [128]. There is no guarantee that model debiased with local data will generalize to the global distribution. The non-i.i.d. data distributions further complicated this problem [88].

Client fairness is tailored to FL and stems from the non-i.i.d. data. Each client sampled the training data from a distinct distribution. In this case, the vanilla FL protocol, *FedAvg*, fails to train a model to fit clients' data distribution. Various methods have been proposed to alleviate this. From the data aspect, References [70, 82, 226] proposed to augment client data to yield an i.i.d. dataset. References [40, 183, 198] proposed to select participant clients to form a more homogeneous distribution. However, their methods did not consider the possible algorithmic fairness issues and may introduce bias to the model by choosing specific clients at a higher probability than others. From the model perspective, training different models for different clients seems a natural solution to the non-i.i.d. challenge. The core idea is to train global data collaboratively and then personalize it to local data distribution.

5 INTERACTIONS BETWEEN PRIVACY AND FAIRNESS

As shown in Figure 3, privacy and fairness are intertwined. On the one hand, *fairness comes at the cost of privacy*. A fair model is trained to perform equally on data from different groups, which incurs overfit problems and consequently increases the privacy risk [21]. On the other hand, *privacy also harms fairness*. Several works [9, 104, 155, 176] witnessed inconsistent reductions in accuracy caused by private mechanisms on classification [48] and generative tasks [56]. Due to the tension between fairness and privacy, researchers often need to make tradeoffs between these two notions. The tradeoff could be increasing privacy protection at the expense of fairness, i.e., adopting relaxed fairness notions instead of exact notions or the opposite way. Table 8 gives various tradeoffs between privacy and fairness. On the basis of the adopted privacy/fairness notions, the tradeoffs can be divided into two categories. The first type sacrifices fairness for solid privacy

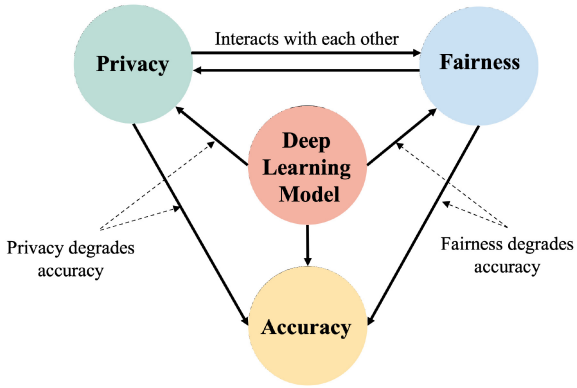


Fig. 3. Privacy, fairness, and accuracy tradeoffs in deep learning: (1) privacy comes at the cost of accuracy; (2) fairness comes at the cost of accuracy; and (3) privacy interacts with fairness [222].

Table 8. Private and Fair Learning

Reference	Privacy notion	Fairness notion	Techniques to achieve		Tradeoff Type
			Privacy	Fairness	
[33]	ϵ-DP	α-Discrimination	Exponential mechanism	Minimize discrimination scores	I
[195]	ϵ-DP	Decision boundary fairness	Functional mechanism	Fairness constraints	I
[137]	ϵ-DP	α-Equal opportunity	Local DP	Post-processing	I
[106]	ϵ-DP	Equal odds & Demographic parity	Class conditional noise	Fairness constraints	I
[36]	ϵ-DP & (ϵ, δ)-DP	Decision boundary fairness	Functional mechanism	Fairness constraints	I
[80]	(ϵ, δ)-DP	α-Equal opportunity	Exponential mechanism & Laplace noise	Fairness constraints	/
[121]	(ϵ, δ)-DP	Equal odds & Demographic parity	DP-SGDA	ERMI regularizer	II
[44]	(ϵ, δ)-DP	Excessive risk gap	DPSGD-Global-Adapt	Gradient correction	II
[175]	(α, ϵ_p)-Rényi DP	Equal odds, Accuracy parity & Demographic parity	DP-SGD	Fairness constraints	II
[100]	/	Equal accuracy	MPC	Fairness constraints	II
[65]	/	Equal opportunity	Proxy attribute	Post-processing	II
[185]	/	Demographic parity	Noisy attribute	Fairness constraints	II
[8]	/	Equal odds	Noisy attribute	Post-processing	II

I: Trade fairness for privacy. Relaxing fairness notions to achieve purely DP.
II: Trade privacy for fairness. Adopting relaxed DP notion to accommodate exact fairness.

protection, i.e., *epsilon*-DP. The second form prioritizes fairness over privacy and employs relaxed DP to accommodate exact fairness.

5.1 Privacy Degrades Fairness

Currently, there are two approaches to privately train a fair model: the cryptography approach and the DP approach. In the first line of works, Veale et al. [180] suggested storing protected characteristics in a trusted third party to protect privacy. The third party will perform discrimination

discovery and incorporate fairness constraints into model-building. Kilbertus et al. [100] relaxed the assumption of a trusted third party and proposed to train a fair model with encrypted sensitive attributes [137]. The cryptography approach is guaranteed to provide comparable performance as the non-private alternative at the cost of communication overhead. The tradeoffs between privacy and efficiency are the main concern in this category.

However, as argued by Reference [80], the cryptographic approach does not guarantee privacy at the inference stage. It only ensures the training data remain private during training and cannot prevent the adversary from inferring training samples from the neural network parameters [169, 224]. On the contrary, DP guarantees a fair model will not leak anything beyond what could be carried out from "population level" correlations. As such, the majority of works focus on learning fair and DP model [9, 33, 80, 195]. Thus, this subsection focuses on DP as the privacy-preserving techniques.

5.1.1 Empirical Findings. The impact of privacy on fairness was initially observed in empirical studies. Bagdasaryan et al. [9] first observed that the reduction in accuracy caused by deep DP models negatively impacts underrepresented subgroups disproportionately. DP-SGD strengthens the model's "bias" toward the most prominent features of the distribution that is being learned. Kuppam et al. [104] reached a similar conclusion when they examined the effects of DP on fairness in three real-world tasks involving sensitive public data. When the noise added by a private algorithm is negligible in relation to the underlying statistics, the costs of adopting a private technique may be minor. When stronger privacy is implemented or when a task entails a small population, significant disparities may emerge. Farrand et al. [48] demonstrated that even minor differences and weak privacy protections could result in disparate outcomes. Ganev et al. [56] shifted the emphasis to generative models and tabular synthetic data. Three DP generative models—PrivBayes [216], DP-WGAN [4], and PATE-GAN [87]—were involved. They witnessed a disparate effect on the accuracy of classifiers trained on synthetic data generated by all generative models. The losses are greater and/or more dispersed for underrepresented groups. Uniyal et al. [179] compared DP-SGD and **Private Aggregation of Teacher Ensembles (PATE)** [146], an alternative DP mechanism for discreetly training a deep neural network, in terms of fairness. They discovered that PATE has a disparate effect, but it is considerably less severe than DP-SGD.

5.1.2 Theoretical Explanations. Several works have attempted to determine the mechanism underlying the well-known relationship between privacy and unfairness. Bagdasaryan et al. [9] associated the impact with the gradient clipping operation in DP-SGD. During training, the model generates larger gradients for samples from underrepresented subgroups; consequently, clipping slows their learning rate. Therefore, the model learns less from the underrepresented subgroups, and its performance on those subgroups is negatively impacted more. Tran et al. [175] conducted an in-depth study into this phenomenon with output perturbation [22] and DP-SGD as the private mechanism. By measuring fairness with *excessive risk gap*, Tran et al. proved that output perturbation mechanisms incur unfairness when the local curvatures of the loss functions of different groups differ substantially. For DP-SGD, Tran et al. found that the clipping bound, the norm of inputs, and the group's distance to the decision boundary collectively contributed to the unfairness raised by DP-SGD. Esipova et al. [44] examined the same issue from the gradient perspective. They proved that the gradient misalignment caused by DP-SGD is the main reason for unfairness. If the clipping operation disproportionately and sufficiently increases the direction error for group a relative to group b, then group a incurs larger excessive risk due to gradient misalignment.

5.1.3 Mitigation Strategies. Diverse methods have been proposed to mitigate the effect of private mechanisms on fairness. Xu et al. [193] proposed DP-SGD-F, a variant of DP-SGD that reduces the divergent impact on different populations. By adaptively designating clipping bounds for each

group, DP-SGD-F achieves a level of privacy proportional to each group's utility-privacy tradeoff. For the group whose clipping bias is greater (due to large gradients), a larger clipping bound is adopted to mitigate for their greater privacy cost. Tran et al. [175] formulated a regularized optimization problem that minimizes empirical loss while satisfying two additional constraints. The first constraint equalizes the averaged non-private and private gradients, while the second constraint penalizes the difference between the local curvatures of distinct groups' loss functions. Esipova et al. [44] modified DP-SGD and developed DP-SGD-Global-Adapt to preserve gradient direction. It is assumed that a hyperparameter Z is the upper bound for most gradients. Gradients less than Z are uniformly scaled, whereas gradients greater than Z are trimmed to Z.

5.2 Fairness Increases Privacy Risk

Fairness, in turn, presents challenges for privacy mechanisms. Chang and Shokri [21] observed an elevated privacy risk for underprivileged subgroups in a fair model. To ensure fairness, the model must perform equally well for all subgroups. However, limited data availability for underprivileged subgroups can lead to overfitting of the training data for unprivileged subgroups in a fair model, thereby increasing the privacy risk. Previous works on fairness-aware machine learning often assume that the sensitive features are reliable and accessible. This assumption unavoidably introduces privacy risks. However, achieving precise notions of fairness, such as demographic parity, becomes unattainable without access to sensitive attributes, specifically the membership information of sensitive groups. To address this issue, several techniques have been proposed to safeguard the privacy of sensitive attributes during the training of fair models.

5.2.1 Training Fair Models with Noisy Representation. Researchers in this field train approximately fair models using noisy sensitive attributes to protect privacy. Gupta et al. [65] substituted protected groups with proxy groups. To achieve fairness, the proxy groups need to align with the true positive group and even overlap with the ground-truth groups. Thus, the fairness guarantee comes at the cost of privacy. Several studies have explored fairness with imperfect group information [8, 89, 106, 185]. Lamy et al. [106] introduced a mutual contaminated model to simulate a noisy distribution with corrupted attributes. Under this framework, they demonstrated that the fairness constraint on the clean distribution is equivalent to a scaled fairness constraint on the noisy distribution. To protect the privacy of sensitive attributes, they added class conditional noise to release the noisy dataset. Awasthi et al. [8] addressed the challenging problem of training a fair model with perturbed sensitive attribute values, where each attribute is independently flipped to its complementary value with probability γ. They identified conditions on the perturbation under which the classifier, denoted as $\hat{Y}$, obtained by Hardt et al.'s method [72], is fairer than the vanilla classifier, denoted as $\tilde{Y}$, trained on accurate attributes. They further provided a formal guarantee of effectiveness under the necessary conditions. Wang et al. [185] trained a fair binary classifier based on a noisy label $\hat{G} \in \{1, \ldots, \hat{m}\}$, i.e., $\hat{G}$ could be *"country of residence"* as a noisy representation of the true group labels $G =$ *"language spoken at home."*

5.2.2 Training Fair Models with DP. Works in this area protect privacy by adding noise to the private characteristic [175]. The tradeoff between privacy and fairness depends on the amount of noise added, with no noise and excessive noise representing the two extremes. In the case of no noise, the model's performance remains unaffected but could lead to information breaches. Conversely, high levels of noise are effective in preserving privacy but can compromise the model's utility. Tran et al. [175] proposed a constrained optimization problem to address both private and fair learning tasks. Their framework ensures (α, ϵ_p)-Rényi DP [132] for the sensitive attributes by solving the constrained problem with DP-SGD. Jagielski et al. [80] extended Agarwal et al.'s

approach [2] by incorporating privacy considerations. They formulated a two-player zero-sum game, played between a "learner" and an "auditor," to derive a fair classifier. Laplacian noise [43] and the exponential mechanism [129] were utilized separately for the "learner" and the "auditor." As a result, the learned model satisfies (ϵ, δ)-DP and achieves equalized odds.

5.3 Fair and Private FL

In centralized machine learning, one entails centralized access to training data (either the true data or noisy data). However, this is invalid in FL, where neither the server nor clients have access to others' data. Therefore, one cannot simply apply centralized fair learning algorithms in FL tasks. This raises a question: *How can we promote algorithmic fairness in FL without accessing clients' data in FL*? Several studies made progress in response to this challenge.

Using a surrogate model to preserve privacy. Padala et al. [143] tried to satisfy both (ϵ, δ)-local DP and demographic fairness through a fair and private FL framework. To circumvent the access restriction, they decomposed the learning into two phases. First, each client learns a fair and accurate model on a local dataset, where the fairness constraint acts as a regularization term in the loss function. Then, every client trains a surrogate model to match the fair predictions from the first model with a DP guarantee. Finally, only the surrogate model is communicated to the server.

Privacy through secure aggregation. Zhang et al. [214] investigated classification problems in FL through multiple goal optimization problems with privacy constraints. The objective is to minimize the accuracy loss and the discrimination risk. To this end, a team Markov game was designed to select participating clients at each communication round. In each round, clients decide whether or not to participate based on the global model's state, which is characterized by bias level and accuracy. Further, a secure aggregation protocol is designed to estimate the global model's status based on polynomial interpolation [95] for privacy concerns. Under this protocol, the server is able to calculate the discrimination status without accessing the local data.

Achieve fairness based on statistics. Gálvez et al. [55] formulated a constrained optimization problem that is solved by the differential multiplier. Local statistics are provided to the server for debiasing the global model. To further protect privacy, client updates are clipped and perturbed by Gaussian noise before being sent to the server. Finally, their solution is able to provide the approximate group fairness notion over multiple attributes and (ϵ, δ)-DP.

Fairness through agnostic learning. Shifts in distribution is one source of bias in FL. The global model is trained on the data of all clients (source distribution), but each client's local data distribution (target distribution) may differ. When deployed to the client, unfavorable outcomes occur. Du et al. [39] proposed treating the client data distribution in an agnostic way. An adversary generates any possible unknown local data distribution to maximize the loss, while the learner aims to optimize the accuracy and fairness.

Calculate fairness violations locally. Chu et al. [31] formulated a constraint optimization problem to learn a fair and private model in FL. Each client locally calculates fairness violations to avoid impinging on the data privacy of any client. Chu et al. [31] further optimized this method by aggregating fairness constraints to better estimate the true fairness violation for all data.

Although some fair FL algorithms do not directly access the training data [31, 39], faithfully sharing the model/gradients in FL could incur privacy leakage risks. The privacy breach could happen during the training or inference stage. The attack could be carried out by either the server or the clients [207]. For example, an honest-but-curious server can launch a reconstruction attack [229] to recover the private data from the gradients uploaded by the victim client. However, the main challenge to training a fair model in FL is restricted data access, e.g., data never leaving local devices, which is an under-investigated topic in FL literature. In the case of the adversary

clients/server in FL, some privacy-preserving techniques, such as DP, can be combined with the aforementioned fair FL approaches to prevent privacy leakage.

5.4 Discussion of Privacy and Fairness Interactions

The complex interactions between privacy and fairness have been thoroughly examined and documented in various studies. These investigations highlight the intricate tradeoffs and challenges that arise when attempting to simultaneously address both privacy and fairness objectives [33].

The impact of privacy and fairness on each other is indeed bilateral. In one scenario, privacy measures can degrade fairness. For instance, in widely used privacy mechanisms like DP-SGD, to protect privacy, the algorithm clips and adds noise to the gradients. However, due to the scarcity of data for certain groups, these modifications can disproportionately affect underrepresented groups, exacerbating unfairness. Therefore, the implementation of DP can inadvertently worsen existing unfairness by disproportionately impacting certain groups.

In another case, fairness can increase privacy risks. To achieve fairness, it may be necessary to collect additional demographic information about users, even if it is irrelevant to the task at hand. This data collection is aimed at guiding modifications to the model, such as addressing inconsistent responses or removing discrimination in statistical models [63, 182, 230]. However, the collection of such sensitive information raises privacy concerns, as it expands the scope of data being collected and potentially increases the risk of privacy breaches.

In the context of FL, the cooperative game between clients and the server adds complexity to the privacy and fairness challenges. FL introduces new privacy attack surfaces, as discussed in Section 3, where potential malicious participants can actively or passively infer the private data of other clients. Consequently, securing private information in FL requires even stronger privacy protection measures compared to the centralized setting. Merely protecting group membership is insufficient to address the privacy risks in FL. Furthermore, the **non-i.i.d. (non-independent and identically distributed)** nature of FL poses another challenge. In a typical FL system, clients' data are sampled from different distributions, leading to data heterogeneity. A model that achieves fairness within the local distribution of each client is not guaranteed to perform unbiasedly on a global scale. The non-i.i.d. issue also introduces potential fairness concerns at the client level, as the performance of the model can vary significantly among clients. It is crucial to address this variation and ensure fairness across all participating clients in FL. The challenge lies in training a fair model in FL without violating the data access restrictions imposed by each client. Finding methods to mitigate the fairness issues arising from the non-i.i.d. nature of the data while respecting the privacy and data access constraints in FL remains a challenging task.

6 OPEN RESEARCH DIRECTIONS

The research community has made fruitful progress in privacy and fairness in FL. However, throughout this survey, we found this field still faces several challenges that need to be solved.

- **Tradeoffs between Privacy and Fairness.** The interaction between privacy and fairness is an under-studied topic. Existing works have focused on exploring the two notions in isolation, either focused on privacy-preserving machine learning [197] or on paradigms that respect fairness [161]. However, as demonstrated by several studies [9, 21, 104], privacy and fairness may compete with each other. In the realm of FL, challenges and opportunities coexist. On the one hand, restricted information and non-i.i.d. distribution complicate the problem settings. On the other hand, the flexibility of the FL paradigm may enable more possible solutions. For instance, the personalized model [103, 173, 227] has been widely used in FL to address statistical challenges by assigning clients personalized models. We may combine

privacy and personalized models to achieve a better tradeoff between privacy, fairness, and utility. Thus, we believe it is worth examining the tradeoffs between privacy and fairness in FL.

- **The Compatibility of Fairness and DP.** We believe it would be worth investigating techniques that simultaneously accommodate fairness and DP. As pointed out in Dwork et al.'s [42] work, given a carefully designed distance metric, it is possible to achieve individual fairness through ϵ-DP. Two characteristics of FL make individual fairness a superior choice over group fairness: (1) Data distribution in FL may vary significantly between clients, and individual fairness is more suitable in such cases. Since it is defined at the sample level, thus, it generates better than group notions when addressing new samples, which may be distinct from those in the training set; (2) The restricted access to information in FL lends itself more to individual fairness, because individual fairness relies on a Lipschitz continual prediction model and does not require access to demographic data. This perfectly fits the FL setting.
- **How Can One Satisfy Fairness at Both the Algorithm and Client Levels in FL?** The majority of studies on fairness in FL focus on promoting fairness at the client level. However, client-level fairness does not necessarily imply algorithmic fairness. Consider a scenario where multiple companies (clients) collaborate to train a credit card approval model. Consumer demographic compositions vary between each company. Although a federated model trained subject to client-level fairness constraints might handle the different companies fairly, the model could still be biased towards sensitive attributes (such as race or educational background). This raises a question: *How can one satisfy fairness at both the algorithm and the client levels while preserving privacy in FL?*

7 CONCLUSION

In this article, we conducted a detailed survey of data privacy and model fairness issues in FL. Uniquely, we also documented the interactions between privacy and fairness from the perspective of tradeoffs. In terms of privacy in FL, we first reviewed privacy attacks in FL. Then, we presented three kinds of privacy-preserving techniques. Regarding fairness, we first analyzed the possible sources of bias and how bias can be introduced in both the client and server sides. Following a review of the notions of fairness adopted in machine learning and those originating from FL, a discussion of the various fairness-aware FL algorithms is presented. The last part of the survey focused on the interactions between privacy and fairness. We identified three relations in the general context and further listed possible solutions to achieve both fair and private FL.

REFERENCES

[1] Annie Abay, Yi Zhou, Nathalie Baracaldo, Shashank Rajamoni, Ebube Chuba, and Heiko Ludwig. 2020. Mitigating bias in federated learning. *arXiv preprint arXiv:2012.02447* (2020).

[2] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *International Conference on Machine Learning*. PMLR, 60–69.

[3] Alekh Agarwal, John Langford, and Chen-Yu Wei. 2020. Federated residual learning. *arXiv preprint arXiv:2003.12880* (2020).

[4] Moustafa Alzantot and Mani Srivastava. Differential privacy synthetic data generation using WGANs, 2019. Retrieved from https://github.com/nesl/nist_differential_privacy_synthetic_data_challenge.

[5] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, LihuaWang, and Shiho Moriai. 2017. Privacy-preserving deep learning: Revisited and enhanced. In *International Conference on Applications and Techniques in Information Security*. Springer, 100–110.

[6] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*. PMLR, 214–223.

[7] Giuseppe Ateniese, Giovanni Felici, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, and Domenico Vitali. 2013. Hacking Smart Machines with Smarter Ones: How to Extract Meaningful Data from Machine Learning Classifiers. DOI: https://doi.org/10.48550/ARXIV.1306.4447

[8] Pranjal Awasthi, Matthäus Kleindessner, and Jamie Morgenstern. 2020. Equalized odds postprocessing under imperfect group information. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1770–1780.
[9] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential privacy has disparate impact on model accuracy. *Adv. Neural Inf. Process. Syst.* 32 (2019), 15479–15488.
[10] Mislav Balunović, Dimitar I. Dimitrov, Robin Staab, and Martin Vechev. 2021. Bayesian framework for gradient leakage. *arXiv preprint arXiv:2111.04706* (2021).
[11] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. 2017. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409* (2017).
[12] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. 2021. Fairness in criminal justice risk assessments: The state of the art. *Sociol. Meth. Res.* 50, 1 (2021), 3–44.
[13] Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam T. Kalai. 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *Adv. Neural Inf. Process. Syst.* 29 (2016), 4349–4357.
[14] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical secure aggregation for privacy-preserving machine learning. In *ACM SIGSAC Conference on Computer and Communications Security*. 1175–1191.
[15] Keith Bonawitz, Fariborz Salehi, Jakub Konečnỳ, Brendan McMahan, and Marco Gruteser. 2019. Federated learning with autotuned communication-efficient secure aggregation. In *53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 1222–1226.
[16] Shikha Bordia and Samuel R. Bowman. 2019. Identifying and reducing gender bias in word-level language models. *arXiv preprint arXiv:1904.03035* (2019).
[17] Christopher Briggs, Zhong Fan, and Péter András. 2020. Federated learning with hierarchical clustering of local updates to improve training on non-IID data. In *International Joint Conference on Neural Networks (IJCNN'20)*. 1–9.
[18] Marc-Etienne Brunet, Colleen Alkalay-Houlihan, Ashton Anderson, and Richard Zemel. 2019. Understanding the origins of bias in word embeddings. In *International Conference on Machine Learning*. PMLR, 803–811.
[19] Toon Calders and Sicco Verwer. 2010. Three naive Bayes approaches for discrimination-free classification. *Data Mining Knowl. Discov.* 21, 2 (2010), 277–292.
[20] Ran Canetti, Aloni Cohen, Nishanth Dikkala, Govind Ramnarayan, Sarah Scheffler, and Adam Smith. 2019. From Soft classifiers to hard decisions: How fair can we be? arXiv:cs.LG/1810.02003.
[21] Hongyan Chang and Reza Shokri. 2021. On the privacy risks of algorithmic fairness. In *IEEE European Symposium on Security and Privacy (EuroS&P'21)*. IEEE, 292–303.
[22] Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. 2011. Differentially private empirical risk minimization. *J. Mach. Learn. Res.* 12, 3 (2011).
[23] Daoyuan Chen, Dawei Gao, Weirui Kuang, Yaliang Li, and Bolin Ding. 2022. pFL-Bench: A comprehensive benchmark for personalized federated learning. *arXiv preprint arXiv:2206.03655* (2022).
[24] Hong-You Chen and Wei-Lun Chao. 2021. On bridging generic and personalized federated learning. *arXiv preprint arXiv:2107.00778* (2021).
[25] Junjie Chen, Wendy Hui Wang, and Xinghua Shi. 2020. Differential privacy protection against membership inference attack on machine learning for genomic data. In *Pacific Symposium (BIOCOMPUTING'21)*. World Scientific, 26–37.
[26] Si Chen, Ruoxi Jia, and Guo-Jun Qi. 2020. Improved techniques for model inversion attacks. *arXiv preprint arXiv:2010.04092* (2020).
[27] Yu Chen, Fang Luo, Tong Li, Tao Xiang, Zheli Liu, and Jin Li. 2020. A training-integrity privacy-preserving federated learning scheme with trusted execution environment. *Inf. Sci.* 522 (2020), 69–79.
[28] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. 2020. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies. *arXiv preprint arXiv:2010.01243* (2020).
[29] Alexandra Chouldechova. 2017. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data* 5, 2 (2017), 153–163.
[30] Alexandra Chouldechova and Aaron Roth. 2018. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810* (2018).
[31] Lingyang Chu, Lanjun Wang, Yanjie Dong, Jian Pei, Zirui Zhou, and Yong Zhang. 2021. FedFair: Training fair models in cross-silo federated learning. *ArXiv* abs/2109.05662 (2021).
[32] Sen Cui, Weishen Pan, Jian Liang, Changshui Zhang, and Fei Wang. 2021. Addressing algorithmic disparity and performance inconsistency in federated learning. *Adv. Neural Inf. Process. Syst.* 34 (2021).
[33] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. 2019. On the compatibility of privacy and fairness. In *27th Conference on User Modeling, Adaptation and Personalization*. 309–315.
[34] Brian d'Alessandro, Cathy O'Neil, and Tom LaGatta. 2017. Conscientious classification: A data scientist's guide to discrimination-aware classification. *Big Data* 5, 2 (2017), 120–134.

[35] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. 2020. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461* (2020).
[36] Jiahao Ding, Xinyue Zhang, Xiaohuan Li, Junyi Wang, Rong Yu, and Miao Pan. 2020. Differentially private and fair classification via calibrated functional mechanism. In *AAAI Conference on Artificial Intelligence*, Vol. 34. 622–629.
[37] Canh T. Dinh, Tung T. Vu, Nguyen H. Tran, Minh N. Dao, and Hongyu Zhang. 2021. FedU: A unified framework for federated multi-task learning with Laplacian regularization. *arXiv preprint arXiv:2102.07148* (2021).
[38] Irit Dinur and Kobbi Nissim. 2003. Revealing information while preserving privacy. In *22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. 202–210.
[39] Wei Du, Depeng Xu, Xintao Wu, and Hanghang Tong. 2021. Fairness-aware agnostic federated learning. In *SIAM International Conference on Data Mining (SDM'21)*. SIAM, 181–189.
[40] M. Duan, D. Liu, X. Chen, R. Liu, Y. Tan, and L. Liang. 2021. Self-balancing federated learning with global imbalanced data in mobile systems. *IEEE Trans. Parallel Distrib. Syst.* 32, 01 (2021), 59–71. DOI: https://doi.org/10.1109/TPDS.2020.3009406
[41] Cynthia Dwork. 2008. Differential privacy: A survey of results. In *International Conference on Theory and Applications of Models of Computation*. Springer, 1–19.
[42] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *3rd Innovations in Theoretical Computer Science Conference*. 214–226.
[43] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, Shai Halevi and Tal Rabin (Eds.). Springer Berlin, 265–284.
[44] Maria S. Esipova, Atiyeh Ashari Ghomi, Yaqiao Luo, and Jesse C. Cresswell. 2022. Disparate impact in differential privacy from gradient misalignment. *arXiv preprint arXiv:2206.07737* (2022).
[45] Yahya H. Ezzeldin, Shen Yan, Chaoyang He, Emilio Ferrara, and Salman Avestimehr. 2021. FairFed: Enabling group fairness in federated learning. *arXiv preprint arXiv:2110.00857* (2021).
[46] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. 2020. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In *Advances in Neural Information Processing Systems*, Vol. 33. 3557–3568.
[47] Lixin Fan, Kam Woh Ng, Ce Ju, Tianyu Zhang, Chang Liu, Chee Seng Chan, and Qiang Yang. 2020. Rethinking privacy preserving deep learning: How to evaluate and thwart privacy attacks. In *Federated Learning*. Springer, 32–50.
[48] Tom Farrand, Fatemehsadat Mireshghallah, Sahib Singh, and Andrew Trask. 2020. Neither private nor fair: Impact of data imbalance on utility and fairness in differential privacy. In *Workshop on Privacy-preserving Machine Learning in Practice*. 15–19.
[49] Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 259–268.
[50] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*. PMLR, 1126–1135.
[51] Joel Escudé Font and Marta R. Costa-Jussà. 2019. Equalizing gender bias in neural machine translation with word embeddings techniques. In *1st Workshop on Gender Bias in Natural Language Processing*. 147–154.
[52] Liam Fowl, Jonas Geiping, Wojtek Czaja, Micah Goldblum, and Tom Goldstein. 2021. Robbing the fed: Directly obtaining private data in federated learning with modified models. *arXiv preprint arXiv:2110.13057* (2021).
[53] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *22nd ACM SIGSAC Conference on Computer and Communications Security*. 1322–1333.
[54] Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. 2014. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *23rd USENIX Security Symposium (USENIX Security'14)*. USENIX Association, 17–32. Retrieved from https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/fredrikson_matthew.
[55] Borja Rodríguez Gálvez, Filip Granqvist, Rogier C. van Dalen, and Matthew Stephen Seigel. 2021. Enforcing fairness in private federated learning via the modified method of differential multipliers. *ArXiv* abs/2109.08604 (2021).
[56] Georgi Ganev, Bristena Oprisanu, and Emiliano De Cristofaro. 2022. Robin Hood and Matthew effects: Differential privacy has disparate impact on synthetic data. In *International Conference on Machine Learning*. PMLR, 6944–6959.
[57] Karan Ganju, Qi Wang, Wei Yang, Carl A. Gunter, and Nikita Borisov. 2018. Property inference attacks on fully connected neural networks using permutation invariant representations. In *ACM SIGSAC Conference on Computer and Communications Security (CCS'18)*. Association for Computing Machinery, New York, NY, 619–633. DOI: https://doi.org/10.1145/3243734.3243834
[58] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. 2020. Inverting gradients—How easy is it to break privacy in federated learning? *arXiv preprint arXiv:2003.14053* (2020).

[59] Robin C. Geyer, Tassilo Klein, and Moin Nabi. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557* (2017).
[60] Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. 2020. An efficient framework for clustered federated learning. *Adv. Neural Inf. Process. Syst.* 33 (2020).
[61] Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P. Friedlander. 2016. Satisfying real-world goals with dataset constraints. In *Conference on Advances in Neural Information Processing Systems*. 2415–2423.
[62] Gene H. Golub and Charles F. Van Loan. 2013. *Matrix Computations*. JHU Press.
[63] Bryce W. Goodman. 2016. A step towards accountable algorithms? Algorithmic discrimination and the European Union general data protection. In *29th Conference on Neural Information Processing Systems (NIPS'16)*. NIPS Foundation.
[64] Zhongshu Gu, Heqing Huang, Jialong Zhang, Dong Su, Hani Jamjoom, Ankita Lamba, Dimitrios Pendarakis, and Ian Molloy. 2019. YerbaBuena: Securing deep learning inference data via enclave-based ternary model partitioning. (2019). arXiv preprint arXiv:1807.00969.
[65] Maya Gupta, Andrew Cotter, Mahdi Milani Fard, and Serena Wang. 2018. Proxy fairness. arXiv:cs.LG/1806.11212.
[66] Otkrist Gupta and Ramesh Raskar. 2018. Distributed learning of deep neural network over multiple agents. *J. Netw. Comput. Applic.* 116 (2018), 1–8.
[67] Samyak Gupta, Yangsibo Huang, Zexuan Zhong, Tianyu Gao, Kai Li, and Danqi Chen. 2022. Recovering private text in federated learning of language models. *arXiv preprint arXiv:2205.08514* (2022).
[68] Farzin Haddadpour, Belhal Karimi, Ping Li, and Xiaoyun Li. 2020. FedSketch: Communication-efficient and private federated learning via sketching. *arXiv preprint arXiv:2008.04975* (2020).
[69] Filip Hanzely and Peter Richtárik. 2021. Federated learning of a mixture of global and local models. arXiv: cs.LG/2002.05516.
[70] Weituo Hao, Mostafa El-Khamy, Jungwon Lee, Jianyi Zhang, Kevin J. Liang, Changyou Chen, and Lawrence Carin Duke. 2021. Towards fair federated learning with zero-shot data augmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21) Workshops*. 3310–3319.
[71] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604* (2018).
[72] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Adv. Neural Inf. Process. Syst.* 29 (2016), 3315–3323.
[73] Ali Hatamizadeh, Hongxu Yin, Holger R. Roth, Wenqi Li, Jan Kautz, Daguang Xu, and Pavlo Molchanov. 2022. GradViT: Gradient inversion of vision transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10021–10030.
[74] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. 2017. Deep models under the GAN: Information leakage from collaborative deep learning. In *ACM SIGSAC Conference on Computer and Communications Security*. 603–618.
[75] Zeou Hu, Kiarash Shaloudegi, Guojun Zhang, and Yaoliang Yu. 2020. FedMGDA+: Federated learning meets multi-objective optimization. arXiv:cs.LG/2006.11489.
[76] Wei Huang, Tianrui Li, Dexian Wang, Shengdong Du, and Junbo Zhang. 2020. Fairness and accuracy in federated learning. arXiv:cs.LG/2012.10069.
[77] Yangsibo Huang, Samyak Gupta, Zhao Song, Kai Li, and Sanjeev Arora. 2021. Evaluating gradient inversion attacks and defenses in federated learning. *Adv. Neural Inf. Process. Syst.* 34 (2021), 7232–7241.
[78] Yangsibo Huang, Zhao Song, Kai Li, and Sanjeev Arora. 2020. InstaHide: Instance-hiding schemes for private distributed learning. In *International Conference on Machine Learning*. PMLR, 4507–4518.
[79] Nick Hynes, Raymond Cheng, and Dawn Song. 2018. Efficient deep learning on multi-source private data. *arXiv preprint arXiv:1807.06689* (2018).
[80] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. 2019. Differentially private fair learning. In *International Conference on Machine Learning*. PMLR, 3000–3008.
[81] Jinwoo Jeon, jaechang Kim, Kangwook Lee, Sewoong Oh, and Jungseul Ok. 2021. Gradient inversion with generative image prior. *Adv. Neural Inf. Process. Syst.* 34 (2021).
[82] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. 2018. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data. arXiv:cs.LG/1811.11479.
[83] Yihan Jiang, Jakub Konečný, Keith Rush, and Sreeram Kannan. 2019. Improving federated learning personalization via model agnostic meta learning. arXiv:cs.LG/1909.12488.
[84] Zhimeng Jiang, Xiaotian Han, Chao Fan, Fan Yang, Ali Mostafavi, and Xia Hu. 2022. Generalized demographic parity for group fairness. In *International Conference on Learning Representations*.
[85] Zhifeng Jiang, Wei Wang, and Yang Liu. 2021. FLASHE: Additively symmetric homomorphic encryption for cross-silo federated learning. *arXiv preprint arXiv:2109.00675* (2021).

[86] Xiao Jin, Pin-Yu Chen, Chia-Yi Hsu, Chia-Mu Yu, and Tianyi Chen. 2021. Catastrophic data leakage in vertical federated learning. *Adv. Neural Inf. Process. Syst.* 34 (2021).
[87] James Jordon, Jinsung Yoon, and Mihaela Van Der Schaar. 2019. PATE-GAN: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*.
[88] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Graham Cormode, Rachel Cummings, and others. 2019. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977* (2019).
[89] Nathan Kallus, Xiaojie Mao, and Angela Zhou. 2020. Assessing algorithmic fairness with unobserved protected class using data combination. arXiv:stat.ML/1906.00285.
[90] Faisal Kamiran and Toon Calders. 2010. Classification with no discrimination by preferential sampling. In *19th Machine Learning Conference*. Citeseer, 1–6.
[91] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowl. Inf. Syst.* 33, 1 (2012), 1–33.
[92] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2012. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 35–50.
[93] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. 2011. Fairness-aware learning through regularization approach. In *IEEE 11th International Conference on Data Mining Workshops*. 643–650. DOI: https://doi.org/10.1109/ICDMW.2011.83
[94] Samhita Kanaparthy, Manisha Padala, Sankarshan Damle, and Sujit Gujar. 2022. Fair federated learning for heterogeneous data. In *5th Joint International Conference on Data Science & Management of Data (CODS-COMAD'22)*. Association for Computing Machinery, New York, NY, 298–299. DOI: https://doi.org/10.1145/3493700.3493750
[95] Ehud Karnin, Jonathan Greene, and Martin Hellman. 1983. On secret sharing systems. *IEEE Trans. Inf. Theor.* 29, 1 (1983), 35–41.
[96] Fumiyuki Kato, Yang Cao, and Masatoshi Yoshikawa. 2022. OLIVE: Oblivious and differentially private federated learning on trusted execution environment. *arXiv preprint arXiv:2202.07165* (2022).
[97] Mohammad Mahdi Khalili, Xueru Zhang, Mahed Abroshan, and Somayeh Sojoudi. 2021. Improving fairness and privacy in selection problems. In *AAAI Conference on Artificial Intelligence*, Vol. 35. 8092–8100.
[98] Haitham Khedr and Yasser Shoukry. 2022. CertiFair: A framework for certified global fairness of neural networks. *arXiv preprint arXiv:2205.09927* (2022).
[99] Mikhail Khodak, Maria-Florina F. Balcan, and Ameet S. Talwalkar. 2019. Adaptive gradient-based meta-learning methods. *Adv. Neural Inf. Process. Syst.* 32 (2019), 5917–5928.
[100] Niki Kilbertus, Adrià Gascón, Matt Kusner, Michael Veale, Krishna Gummadi, and Adrian Weller. 2018. Blind justice: Fairness with encrypted sensitive attributes. In *International Conference on Machine Learning*. PMLR, 2630–2639.
[101] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. Overcoming catastrophic forgetting in neural networks. *Proc. Nat. Acad. Sci.* 114, 13 (2017), 3521–3526.
[102] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2017. Inherent trade-offs in the fair determination of risk scores. In *8th Innovations in Theoretical Computer Science Conference (ITCS'17)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
[103] Viraj Kulkarni, Milind Kulkarni, and Aniruddha Pant. 2020. Survey of personalization techniques for federated learning. In *4th World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4'20)*. IEEE, 794–797.
[104] Satya Kuppam, Ryan McKenna, David Pujol, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. 2020. Fair decision making using privacy-protected data. In *Conference on Fairness, Accountability, and Transparency*.
[105] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2018. Counterfactual fairness. arXiv: stat.ML/1703.06856.
[106] Alex Lamy, Ziyuan Zhong, Aditya K. Menon, and Nakul Verma. 2019. Noise-tolerant fair classification. *Adv. Neural Inf. Process. Syst.* 32 (2019).
[107] Tian Lan, David Kao, Mung Chiang, and Ashutosh Sabharwal. 2010. *An Axiomatic Theory of Fairness in Network Resource Allocation*. IEEE.
[108] Klas Leino and Matt Fredrikson. 2020. Stolen memories: Leveraging model memorization for calibrated White-Box membership inference. In *29th USENIX Security Symposium (USENIX Security'20)*. 1605–1622.
[109] Daliang Li and Junpu Wang. 2019. FedMD: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581* (2019).
[110] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. 2021. Ditto: Fair and robust federated learning through personalization. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of*

Machine Learning Research), Marina Meila and Tong Zhang (Eds.), Vol. 139. PMLR, 6357–6368. Retrieved from https://proceedings.mlr.press/v139/li21h.html.

[111] Tian Li, Zaoxing Liu, Vyas Sekar, and Virginia Smith. 2019. Privacy for free: Communication-efficient learning with differential privacy using sketches. *arXiv preprint arXiv:1911.00972* (2019).

[112] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proc. Mach. Learn. Syst.* 2 (2020), 429–450.

[113] Tian Li, Maziar Sanjabi, Ahmad Beirami, and Virginia Smith. 2019. Fair resource allocation in federated learning. *arXiv preprint arXiv:1905.10497* (2019).

[114] Zhize Li, Haoyu Zhao, Boyue Li, and Yuejie Chi. 2022. SoteriaFL: A unified framework for private federated learning with communication compression. *arXiv preprint arXiv:2206.09888* (2022).

[115] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B. Allen, Randy P. Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2020. Think locally, act globally: Federated learning with local and global representations. arXiv:cs.LG/2001.01523.

[116] Shiyun Lin, Yuze Han, Xiang Li, and Zhihua Zhang. 2020. Personalized federated learning towards communication efficiency, robustness and fairness. *Adv. Neural Inf. Process. Syst.* 35 (2020).

[117] Junxu Liu and Xiaofeng Meng. 2020. Survey on privacy-preserving machine learning. *J. Comput. Res. Devel.* 57, 2 (2020), 346.

[118] Yang Liu, Yan Kang, Chaoping Xing, Tianjian Chen, and Qiang Yang. 2020. A secure federated transfer learning framework. *IEEE Intell. Syst.* 35, 4 (2020), 70–82.

[119] Pranay K. Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R. Varshney, and Ruchir Puri. 2019. Bias mitigation post-processing for individual and group fairness. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'19)*. 2847–2851. DOI: https://doi.org/10.1109/ICASSP.2019.8682620

[120] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. 2016. The variational fair autoencoder. In *International Conference on Learning Representations*.

[121] Andrew Lowy, Devansh Gupta, and Meisam Razaviyayn. 2023. Stochastic differentially private and fair learning. In *International Conference on Learning Representations*.

[122] Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. 2021. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. *Adv. Neural Inf. Process. Syst.* 34 (2021), 5972–5984.

[123] Lingjuan Lyu, Han Yu, and Qiang Yang. 2020. Threats to federated learning: A survey. *arXiv preprint arXiv:2003.02133* (2020).

[124] Xindi Ma, Baopu Li, Qi Jiang, Yimin Chen, Sheng Gao, and Jianfeng Ma. 2021. NOSnoop: An effective collaborative meta-learning scheme against property inference attack. *IEEE Internet Things J.* 9, 9 (2021), 6778–6789.

[125] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. 2020. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619* (2020).

[126] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. 2020. Minimax Pareto fairness: A multi objective perspective. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Hal Daumé III and Aarti Singh (Eds.), Vol. 119. PMLR, 6755–6764.

[127] Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. *arXiv preprint arXiv:1903.10561* (2019).

[128] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*. PMLR, 1273–1282.

[129] Frank McSherry and Kunal Talwar. 2007. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*. IEEE, 94–103.

[130] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Comput. Surv.* 54, 6 (2021), 1–35.

[131] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *IEEE Symposium on Security and Privacy (SP'19)*. IEEE, 691–706.

[132] Ilya Mironov. 2017. Rényi differential privacy. In *IEEE 30th Computer Security Foundations Symposium (CSF'17)*. IEEE, 263–275.

[133] Fan Mo, Ali Shahin Shamsabadi, Kleomenis Katevas, Soteris Demetriou, Ilias Leontiadis, Andrea Cavallaro, and Hamed Haddadi. 2020. DarkneTZ: Towards model privacy at the edge using trusted execution environments. In *18th International Conference on Mobile Systems, Applications, and Services*. 161–174.

[134] Jeonghoon Mo and Jean Walrand. 2000. Fair end-to-end window-based congestion control. *IEEE/ACM Trans. Netw.* 8, 5 (2000), 556–567.

[135] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. 2019. Agnostic federated learning. In *International Conference on Machine Learning*. PMLR, 4615–4625.

[136] Arup Mondal, Yash More, Prashanthi Ramachandran, Priyam Panda, Harpreet Virk, and Debayan Gupta. 2022. SCOTCH: An efficient secure computation framework for secure aggregation. *arXiv preprint arXiv:2201.07730* (2022).
[137] Hussein Mozannar, Mesrob Ohannessian, and Nathan Srebro. 2020. Fair learning with private demographic data. In *International Conference on Machine Learning*. PMLR, 7066–7075.
[138] Lokesh Nagalapatti and Ramasuri Narayanam. 2021. Game of gradients: Mitigating irrelevant clients in federated learning. *Proc. AAAI Conf. Artif. Intell.* 35, 10 (5 2021), 9046–9054. Retrieved from https://ojs.aaai.org/index.php/AAAI/article/view/17093.
[139] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *IEEE Symposium on Security and Privacy (SP'19)*. 739–753. DOI: https://doi.org/10.1109/SP.2019.00065
[140] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On first-order meta-learning algorithms. arXiv: cs.LG/1803.02999.
[141] Takayuki Nishio and Ryo Yonetani. 2019. Client selection for federated learning with heterogeneous resources in mobile edge. In *IEEE International Conference on Communications (ICC'19)*. IEEE, 1–7.
[142] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. 2019. Social data: Biases, methodological pitfalls, and ethical boundaries. *Front. Big Data* 2 (2019), 13.
[143] Manisha Padala, Sankarshan Damle, and Sujit Gujar. 2021. Federated learning meets fairness and differential privacy. In *International Conference on Neural Information Processing*. Springer, 692–699.
[144] Xudong Pan, Mi Zhang, Yifan Yan, Jiaming Zhu, and Min Yang. 2020. Exploring the security boundary of data reconstruction via neuron exclusivity analysis. *arXiv preprint arXiv:2010.13356* (2020).
[145] Afroditi Papadaki, Natalia Martinez, Martin Bertran, Guillermo Sapiro, and Miguel Rodrigues. 2021. Federating for learning group fair models. arXiv:cs.LG/2110.01999.
[146] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2016. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755* (2016).
[147] Jaehyoung Park and Hyuk Lim. 2022. Privacy-preserving federated learning using homomorphic encryption. *Appl. Sci.* 12, 2 (2022), 734.
[148] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang, and Shiho Moriai. 2018. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Trans. Inf. Forens. Secur.* 13 (2018), 1333–1345.
[149] Le Trieu Phong and Tran Thi Phuong. 2019. Privacy-preserving deep learning via weight transmission. *IEEE Trans. Inf. Forens. Secur.* 14, 11 (2019), 3003–3015.
[150] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q. Weinberger. 2017. On fairness and calibration. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2017/file/b8b9c74ac526fffbeb2d39ab038d1cd7-Paper.pdf.
[151] Jia Qian and Lars Kai Hansen. 2020. What can we learn from gradients? arXiv preprint arXiv:2010.15718.
[152] Hanchi Ren, Jingjing Deng, and Xianghua Xie. 2022. GRNN: Generative Regression Neural Network—A data leakage attack for federated learning. *ACM Trans. Intell. Syst. Technol.* (12 2022). DOI: https://doi.org/10.1145/3510032
[153] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. 2020. Updates-Leak: Data set inference and reconstruction attacks in online learning. In *29th USENIX Security Symposium (USENIX Security'20)*. 1291–1308.
[154] Tiago Salvador, Stephanie Cairns, Vikram Voleti, Noah Marshall, and Adam Oberman. 2021. FairCal: Fairness calibration for face verification. *arXiv preprint arXiv:2106.03761* (2021).
[155] Amartya Sanyal, Yaxi Hu, and Fanny Yang. 2022. How unfair is private learning? In *Uncertainty in Artificial Intelligence*. PMLR, 1738–1748.
[156] Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *57th Annual Meeting of the Association for Computational Linguistics*. 1668–1678.
[157] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. 2020. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Trans. Neural Netw. Learn. Syst.* 32, 8 (2020), 3710–3722.
[158] Daniel Scheliga, Patrick Mäder, and Marco Seeland. 2022. PRECODE—A generic model extension to prevent deep gradient leakage. In *IEEE/CVF Winter Conference on Applications of Computer Vision*. 1849–1858.
[159] Adi Shamir. 1979. How to share a secret. *Commun. ACM* 22, 11 (1979), 612–613.
[160] Jiawei Shao, Yuchang Sun, Songze Li, and Jun Zhang. 2022. DReS-FL: Dropout-resilient secure federated learning for non-iid clients via secret data sharing. *arXiv preprint arXiv:2210.02680* (2022).
[161] Yuxin Shi, Han Yu, and Cyril Leung. 2021. A survey of fairness-aware federated learning. *arXiv preprint arXiv:2111.01872* (2021).

[162] Jinmyeong Shin, Seok-Hwan Choi, and Yoon-Ho Choi. 2021. Is homomorphic encryption-based deep learning secure enough? *Sensors* 21, 23 (2021), 7806.
[163] Neta Shoham, Tomer Avidor, Aviv Keren, Nadav Israel, Daniel Benditkis, Liron Mor-Yosef, and Itai Zeitak. 2019. Overcoming forgetting in federated learning on non-iid data. *arXiv preprint arXiv:1910.07796* (2019).
[164] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *22nd ACM SIGSAC Conference on Computer and Communications Security*. 1310–1321.
[165] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (SP'17)*. IEEE, 3–18.
[166] Karan Singhal, Hakim Sidahmed, Zachary Garrett, Shanshan Wu, Keith Rush, and Sushant Prakash. 2021. Federated reconstruction: Partially local federated learning. arXiv:cs.LG/2102.03448.
[167] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S. Talwalkar. 2017. Federated multi-task learning. In *Conference on Neural Information Processing Systems (NIPS'17)*.
[168] Jinhyun So, Başak Güler, and A. Salman Avestimehr. 2021. Turbo-aggregate: Breaking the quadratic aggregation barrier in secure federated learning. *IEEE J. Select. Areas Inf. Theor.* 2, 1 (2021), 479–489.
[169] Congzheng Song, Thomas Ristenpart, and Vitaly Shmatikov. 2017. Machine learning models that remember too much. In *ACM SIGSAC Conference on Computer and Communications Security*. 587–601.
[170] Mengkai Song, Zhibo Wang, Zhifei Zhang, Yang Song, Qian Wang, Ju Ren, and Hairong Qi. 2020. Analyzing user-level privacy attack against federated learning. *IEEE J. Select. Areas Commun.* 38, 10 (2020), 2430–2444.
[171] Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. 2013. Stochastic gradient descent with differentially private updates. In *IEEE Global Conference on Signal and Information Processing*. IEEE, 245–248.
[172] Ruoyu Sun, Tiantian Fang, and Alex Schwing. 2020. Towards a better global loss landscape of GANs. DOI: https://doi.org/10.48550/ARXIV.2011.04926
[173] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. 2021. Towards personalized federated learning. *arXiv preprint arXiv:2103.00710* (2021).
[174] Florian Tramer and Dan Boneh. 2018. Slalom: Fast, verifiable and private execution of neural networks in trusted hardware. *arXiv preprint arXiv:1806.03287* (2018).
[175] Cuong Tran, Ferdinando Fioretto, and Pascal Van Hentenryck. 2021. Differentially private and fair deep learning: A Lagrangian dual approach. In *AAAI Conference on Artificial Intelligence*, Vol. 35. 9932–9939.
[176] Cuong Tran, Ferdinando Fioretto, Pascal Van Hentenryck, and Zhiyan Yao. 2021. Decision making with differential privacy under a fairness lens. In *International Joint Conference on Artificial Intelligence (IJCAI'21)*.
[177] 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 *12th ACM Workshop on Artificial Intelligence and Security*. 1–11.
[178] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu, and Wenqi Wei. 2019. Demystifying membership inference attacks in machine learning as a service. *IEEE Trans. Serv. Comput.* 14, 6 (2019), 2073–2089. DOI: https://doi.org/10.1109/TSC.2019.2897554
[179] Archit Uniyal, Rakshit Naidu, Sasikanth Kotti, Sahib Singh, Patrik Joslin Kenfack, Fatemehsadat Mireshghallah, and Andrew Trask. 2021. DP-SGD vs PATE: Which has less disparate impact on model accuracy? *arXiv preprint arXiv:2106.12576* (2021).
[180] Michael Veale and Reuben Binns. 2017. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data Soc.* 4, 2 (2017), 205395171774353O.
[181] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *International Workshop on Software Fairness (FairWare'18)*. Association for Computing Machinery, New York, NY, 1–7. DOI: https://doi.org/10.1145/3194770.3194776
[182] Paul Voigt and Axel Von dem Bussche. 2017. The EU General Data Protection Regulation (GDPR). *A Practical Guide, 1st Ed., Cham: Springer International Publishing* 10 (2017), 3152676.
[183] Hao Wang, Zakhary Kaplan, Di Niu, and Baochun Li. 2020. Optimizing federated learning on non-IID data with reinforcement learning. In *IEEE Conference on Computer Communications*. 1698–1707. DOI: https://doi.org/10.1109/INFOCOM41043.2020.9155494
[184] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. 2019. Eavesdrop the composition proportion of training labels in federated learning. DOI: https://doi.org/10.48550/ARXIV.1910.06044
[185] Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Michael Jordan. 2020. Robust optimization for fairness with noisy protected groups. *Adv. Neural Inf. Process. Syst.* 33 (2020), 5190–5203.
[186] Yijue Wang, Jieren Deng, Dan Guo, Chenghong Wang, Xianrui Meng, Hang Liu, Caiwen Ding, and Sanguthevar Rajasekaran. 2020. SAPAG: A self-adaptive privacy attack from gradients. *arXiv preprint arXiv:2009.06228* (2020).
[187] Zhibo Wang, Yuting Huang, Mengkai Song, Libing Wu, Feng Xue, and Kui Ren. 2022. Poisoning-assisted property inference attack against federated learning. *IEEE Trans. Depend. Secure Comput.* 1 (2022), 1–1.
[188] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. 2019. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *IEEE Conference on Computer Communications*. IEEE, 2512–2520.

[189] Kang Wei, Jun Li, Ming Ding, Chuan Ma, Howard H. Yang, Farhad Farokhi, Shi Jin, Tony Q. S. Quek, and H. Vincent Poor. 2020. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Trans. Inf. Forens. Secur.* 15 (2020), 3454–3469.

[190] Wenqi Wei and Ling Liu. 2021. Gradient leakage attack resilient deep learning. *IEEE Trans. Inf. Forens. Secur.* (2021).

[191] Wenqi Wei, Ling Liu, Margaret Loper, Ka-Ho Chow, Mehmet Emre Gursoy, Stacey Truex, and Yanzhao Wu. 2020. A framework for evaluating gradient leakage attacks in federated learning. *arXiv preprint arXiv:2004.10397* (2020).

[192] Yuezhou Wu, Yan Kang, Jiahuan Luo, Yuanqin He, and Qiang Yang. 2021. FedCG: Leverage conditional GAN for protecting privacy and maintaining competitive performance in federated learning. *arXiv preprint arXiv:2111.08211* (2021).

[193] Depeng Xu, Wei Du, and Xintao Wu. 2021. Removing disparate impact on model accuracy in differentially private stochastic gradient descent. In *27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD'21)*. Association for Computing Machinery, New York, NY, 1924–1932. DOI: https://doi.org/10.1145/3447548.3467268

[194] Depeng Xu, Shuhan Yuan, and Xintao Wu. 2019. Achieving differential privacy and fairness in logistic regression. In *World Wide Web Conference (WWW'19)*. Association for Computing Machinery, New York, NY, 594–599. DOI: https://doi.org/10.1145/3308560.3317584

[195] Depeng Xu, Shuhan Yuan, and Xintao Wu. 2019. Achieving differential privacy and fairness in logistic regression. In *World Wide Web Conference*. 594–599.

[196] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. 2018. FairGAN: Fairness-aware generative adversarial networks. In *IEEE International Conference on Big Data (Big Data'18)*. IEEE, 570–575.

[197] Runhua Xu, Nathalie Baracaldo, and James Joshi. 2021. Privacy-preserving machine learning: Methods, challenges and directions. *arXiv preprint arXiv:2108.04417* (2021).

[198] Miao Yang, Akitanoshou Wong, Hongbin Zhu, Haifeng Wang, and Hua Qian. 2020. Federated learning with class imbalance reduction. *arXiv preprint arXiv:2011.11266* (2020).

[199] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.* 10, 2 (2019), 1–19.

[200] Wenzhuo Yang, Yipeng Zhou, Miao Hu, Di Wu, Xi Zheng, Jessie Hui Wang, Song Guo, and Chao Li. 2021. Gain without pain: Offsetting DP-injected noises stealthily in cross-device federated learning. *IEEE Internet Things J.* 9, 22 (2021), 22147–22157.

[201] Xue Yang, Yan Feng, Weijun Fang, Jun Shao, Xiaohu Tang, Shu-Tao Xia, and Rongxing Lu. 2020. An accuracy-lossless perturbation method for defending privacy attacks in federated learning. *arXiv preprint arXiv:2002.09843* (2020).

[202] Ziqi Yang, Ee-Chien Chang, and Zhenkai Liang. 2019. Adversarial neural network inversion via auxiliary knowledge alignment. *arXiv preprint arXiv:1902.08552* (2019).

[203] Andrew C. Yao. 1982. Protocols for secure computations. In *23rd Annual Symposium on Foundations of Computer Science (SFCS'82)*. IEEE, 160–164.

[204] Xin Yao and Lifeng Sun. 2020. Continual local training for better initialization of federated models. In *IEEE International Conference on Image Processing (ICIP'20)*. IEEE, 1736–1740.

[205] Xun Yi, Russell Paulet, and Elisa Bertino. 2014. Homomorphic encryption. In *Homomorphic Encryption and Applications*. Springer, 27–46.

[206] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M. Alvarez, Jan Kautz, and Pavlo Molchanov. 2021. See through gradients: Image batch recovery via GradInversion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16337–16346.

[207] Xuefei Yin, Yanming Zhu, and Jiankun Hu. 2021. A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions. *ACM Comput. Surv.* 54, 6 (2021), 1–36.

[208] Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. 2020. A fairness-aware incentive scheme for federated learning. In *AAAI/ACM Conference on AI, Ethics, and Society*. 393–399.

[209] Xubo Yue, Maher Nouiehed, and Raed Al Kontar. 2021. GIFAIR-FL: An approach for group and individual fairness in federated learning. arXiv:cs.LG/2108.02741.

[210] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P. Gummadi. 2017. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*. PMLR, 962–970.

[211] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *International Conference on Machine Learning*. PMLR, 325–333.

[212] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *AAAI/ACM Conference on AI, Ethics, and Society*. 335–340.

[213] Chengliang Zhang, Suyi Li, Junzhe Xia, Wei Wang, Feng Yan, and Yang Liu. 2020. BatchCrypt: Efficient homomorphic encryption for Cross-Silo federated learning. In *USENIX Annual Technical Conference (USENIX ATC'20)*. 493–506.

[214] Daniel Yue Zhang, Ziyi Kou, and Dong Wang. 2020. FairFLl: A fair federated learning approach to reducing demographic bias in privacy-sensitive classification models. In *IEEE International Conference on Big Data (Big Data'20)*. IEEE, 1051–1060.

[215] Jiale Zhang, Bing Chen, Shui Yu, and Hai Deng. 2019. PEFL: A privacy-enhanced federated learning scheme for big data analytics. In *IEEE Global Communications Conference (GLOBECOM'19)*. IEEE, 1–6.
[216] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2017. PRIVBAYES: Private data release via Bayesian networks. *ACM Trans. Datab. Syst.* 42, 4 (2017), 1–41.
[217] Jingwen Zhang, Jiale Zhang, Junjun Chen, and Shui Yu. 2020. GAN enhanced membership inference: A passive local attack in federated learning. In *IEEE International Conference on Communications (ICC'20)*. 1–6. DOI: https://doi.org/10.1109/ICC40277.2020.9148790
[218] Jingwen Zhang, Jiale Zhang, Junjun Chen, and Shui Yu. 2020. GAN enhanced membership inference: A passive local attack in federated learning. In *IEEE International Conference on Communications (ICC'20)*. IEEE, 1–6.
[219] L. Zhang, Y. Wu, and X. Wu. 2019. Fairness-aware classification: Criterion convexity and bounds. In *AAAI Conference on Artificial Intelligence*.
[220] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. 2020. Personalized federated learning with first order model optimization. In *International Conference on Learning Representations*.
[221] Mengjiao Zhang and Shusen Wang. 2021. Matrix sketching for secure collaborative machine learning. In *International Conference on Machine Learning*. PMLR, 12589–12599.
[222] Tao Zhang, Tianqing Zhu, Kun Gao, Wanlei Zhou, and S. Yu Philip. 2021. Balancing learning model privacy, fairness, and accuracy with early stopping criteria. *IEEE Trans. Neural Netw. Learn. Syst.* (2021).
[223] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. 2020. The secret revealer: Generative model-inversion attacks against deep neural networks. arXiv:cs.LG/1911.07135.
[224] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. 2020. IDLG: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610* (2020).
[225] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. *arXiv preprint arXiv:1804.06876* (2018).
[226] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. 2018. Federated learning with non-iid data. *arXiv preprint arXiv:1806.00582* (2018).
[227] Hangyu Zhu, Jinjin Xu, Shiqing Liu, and Yaochu Jin. 2021. Federated learning on non-IID data: A survey. *Neurocomputing* 465 (2021), 371–390.
[228] Junyi Zhu and Matthew Blaschko. 2020. R-GAP: Recursive gradient attack on privacy. *arXiv preprint arXiv:2010.07733* (2020).
[229] Ligeng Zhu, Zhijian Liu, and Song Han. 2019. Deep leakage from gradients. *Adv. Neural Inf. Process. Syst.* 32 (2019).
[230] Indrė Žliobaitė and B. H. M. Custers. 2016. Using sensitive personal data may be necessary for avoiding discrimination in data-driven decision models. *Artif. Intell. Law* 24 (2016), 183–201.

Received 27 April 2022; revised 16 June 2023; accepted 22 June 2023