Privacy Inference Attack and Defense in Centralized and Federated Learning: A Comprehensive Survey

Bosen Rao, Jiale Zhang, *Member, IEEE,* Di Wu, *Member, IEEE,* Chengcheng Zhu, Xiaobing Sun, *Member, IEEE,* Bing Chen, *Member, IEEE*

***Abstract*—The emergence of new machine learning methods has led to their widespread application across various domains, significantly advancing the field of artificial intelligence. However, the process of training and inferring machine learning models relies on vast amounts of data, which often includes sensitive private information. Consequently, the privacy and security of machine learning have encountered significant challenges. Several studies have demonstrated the vulnerability of machine learning to privacy inference attacks, but they often focus on specific scenarios, leaving a gap in understanding the broader picture. We provide a comprehensive review of privacy attacks in machine learning, focusing on two scenarios: centralized learning and federated learning. This paper begins by presenting the architectures of both centralized learning and federated learning, along with their respective application scenarios. It then conducts a comprehensive review and categorization of related inference attacks, providing a detailed analysis of the different stages involved in these attacks. Moreover, the paper thoroughly describes and compares the existing defense methods. Finally, the paper concludes by highlighting open questions and potential future research directions, aiming to contribute to the ongoing competition between privacy attackers and defenders.**

***Impact Statement*—Machine learning models have consistently shown impressive performance across numerous domains. However, the mounting concern about privacy issues in machine learning has led to an increased emphasis on this topic. This paper primarily aims to conduct a thorough review of two scenarios: centralized learning and federated learning, summarizing potential inference attacks at different stages of the machine learning model lifecycle. In addition, we provide a comprehensive introduction and comparison of existing defenses. The goal of this paper is to contribute to the ongoing discussion on privacy inference attack and defense by providing a thorough analysis of inference attacks and their associated protective measures.**

***Index Terms*—Privacy inference attack, Privacy defense, Centralized and Federated learning, Machine learning security**

I. Introduction

MACHINE learning has proven to be highly effective in various domains, such as image recognition [1], natural language processing [2], graph data applications [3], computer vision [4], email filtering [5], and more. In the traditional centralized learning approach, data is typically centralized on Cloud Virtual Machines to build powerful inference models. However, due to increasing concerns about privacy and the implementation of privacy laws, data owners have become more cautious about sharing their data. As a result, this has led to the emergence of data silos [6], where data is siloed and not easily shared. Furthermore, the transmission of large amounts of data to the cloud can lead to data congestion and significant propagation delays.

To effectively address this challenge, federated learning was proposed by Google in 2016 [7]. Federated learning has emerged as a promising approach to train models on distributed datasets while preserving data privacy. Through the establishment of a common model that is shared among all participating devices or servers, federated learning enables model training to take place locally on each device without the need to transmit raw data to a central server. By establishing a common model, all participants can benefit from collective knowledge while keeping their data secure and private. Federated learning is particularly suitable for privacy-sensitive scenarios, such as smart healthcare [8], banking transactions [9], and keyboard prediction [10].

Indeed, despite its potential benefits, federated learning is not immune to security and privacy challenges. Recent research [11, 12] has highlighted several concerns that may compromise the privacy guarantees of federated learning. Firstly, potential privacy leaks may occur not only from raw data but also through gradient communicatios nduring the entire training process [13, 14], possibly leading to even more profound privacy breaches [15]. Secondly, the participation of numerous actors in federated learning makes it challenging to verify the credibility of all participants. There is a risk of adversarial actors who may infer sensitive information about others, manipulate global parameter aggregations, or corrupt the global model. Additionally, malicious servers in federated learning could exploit single updates over time to infer sensitive information, tamper with the training process, or manipulate participants' perception of global parameters.

Compared to adversarial attacks that manipulate the model, privacy inference attacks solely aim to extract private information without disrupting the model's normal training process. This characteristic renders them more covert and substantially increases the associated risks. To provide a comprehensive overview of these attacks, we categorize both centralized and federated learning into three key stages: data acquisition, training, and prediction. To comprehensively understand these risks, we will analyze each stage from both the attacker's and

Jiale Zhang is the corresponding author.

Bosen Rao, Jiale Zhang, Chengcheng Zhu and Xiaobing Sun are with the School of Information Engineering, Yangzhou University, Yangzhou, China, 225127 (e-mail: MX120230573@stu.yzu.edu.cn; jialezhang@yzu.edu.cn; MX120220554@stu.yzu.edu.cn; xbsun@yzu.edu.cn).

Di Wu is with the School of Mathematics, Physics and Computing, University of Southern Queensland, Toowoomba, 4350, Australia (e-mail: di.wu@unisq.edu.au).

Bing Chen is with College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 211106 (e-mail: cb_china@nuaa.edu.cn).

defender's perspectives to identify potential privacy vulnerabilities. To comprehensively understand privacy attacks, we can categorize them into three main types: 1) Membership inference attack: This attack aims to infer whether a specific personal data record is present in the training dataset. It seeks to identify if an individual's data was used in the training process. 2) Property inference attack: The goal of this attack is to infer certain properties or attributes that are present in the training dataset. By analyzing the model's behavior or predictions, an attacker attempts to deduce sensitive information about individuals. 3) Model extraction attack: This attack aims to recover the original model parameters or functions by exploiting the API interface of the model. The attacker may construct a model that closely resembles the original one or extract valuable intellectual property from the model. To counter these privacy attacks, various defense methods have been developed. These methods include differential privacy, adversarial machine learning, and encryption techniques. Their primary objectives are to protect data privacy and mitigate information leakage from the model or its output.

Until now, researchers have conducted separate reviews of privacy issues in machine learning and security and privacy protection in federated learning. However, there is still a lack of a systematic and comprehensive review that encompasses privacy reasoning across various scenarios. This paper aims to fill this gap by focusing on privacy inference attack threats and related defense measures in both the centralized learning scenarios and the federated learning scenarios. The paper will provide a thorough examination of previous works and research status, systematically classifying and analyzing attack and defense methods. It will offer an extensive overview of privacy inference attacks and defense technologies. By covering both centralized learning and federated learning scenarios, this paper aims to provide a comprehensive understanding of privacy-related challenges and solutions in machine learning. Through this systematic review, researchers and practitioners will gain valuable insights into the landscape of privacy inference attacks and defense techniques.

The remainder of this survey is organized as follows: Section 2 provides an introduction to the basic concepts and applications of both centralized learning and federated learning. In Section 3, we conduct a comprehensive literature review of privacy attacks. We delve into the implementation stages of several common attacks and discuss the advancements made in research on these attacks. This section offers a detailed overview of the current state of privacy attacks in machine learning. Section 4 analyzes defense methods based on the existing attack methods and research results. We explore defense strategies and techniques that can be applied at different stages of the machine learning process. In Section 5, we discuss the challenges associated with privacy inference attacks and highlight areas for future research. Finally, Section 6 presents the conclusions of this survey.

II. Centralized and Federated Learning

In the early stages of machine learning, when the amount of data was relatively small, the predominant form of learning was centralized learning. In centralized learning, data is gathered and stored on a single server, and the learning model is trained using this centralized data source. However, with the emergence of big data and advancements in computing power, the limitations of centralized learning became apparent. Training large-scale models with vast amounts of data on a single server became impractical due to computational constraints. To address this challenge, centralized learning techniques were introduced. Centralized learning involves distributing the training process across multiple computing nodes or servers. More recently, the focus on privacy has given rise to a specific form of centralized learning known as federated learning. In this summary, we aim to provide an overview of the fundamental concepts of centralized learning and federated learning, along with their respective application scenarios.

TABLE I
NOTATIONS

Symbol	Meaning
F	data owner
D	respective data
M	model
V	the accuracy of model
P	participant

A. Centralized Learning

Centralized learning refers to a machine learning paradigm where both the data sets and the model training process are centralized at a single location or server. In this approach, data is gathered and stored on a central server, which is responsible for training the model. The participants' role in centralized learning is primarily to provide their data to the central server, without direct involvement in the model training process. It is worth noting that in centralized learning, a single participant assumes the responsibility for managing all training processes and resources. This approach is commonly used in traditional machine learning models.

B. Federated Learning

In 2016, Google introduced the concept of federated learning [7], which seeks to train models on distributed datasets while preserving the privacy of individual data. Building upon this concept, Yang et al. [6] proposed a comprehensive and secure federated learning framework that extends the concept of federated learning to the general concept of all privacy-preserving decentralized collaborative machine learning techniques. Their framework provides a robust foundation for enabling collaborative machine learning while ensuring the protection of sensitive data. Table I lists the common notations used in the paper.

Federated learning involves a scenario where there are N data owners, denoted as $\{F_1, ..., F_N\}$, each possessing their respective datasets $\{D_1, ..., D_N\}$. The crucial aspect is that these data owners do not share their datasets with others. Initially, each participant downloads the initial model M_{fed}^0 to their local device and trains it using their dataset D_i. Through

iterative updates, the final model M_{fed} is generated, which has a minimal difference in performance compared to the effect V_{sum} obtained by directly integrating all D_i for model training, as:

$$|V_{Fed} - V_{sum}| < \delta \tag{1}$$

Where δ is an arbitrarily small positive number, this approach allows participants to achieve the same training effect as if they had directly integrated their local private data for training. It ensures that participants can contribute to the model training without compromising the privacy of their data, thereby fully protecting the data privacy of all parties involved. The basic framework of the federated learning system, along with the specific details of each step, is illustrated in Fig. 1. This figure provides a visual representation of the overall process involved in federated learning.

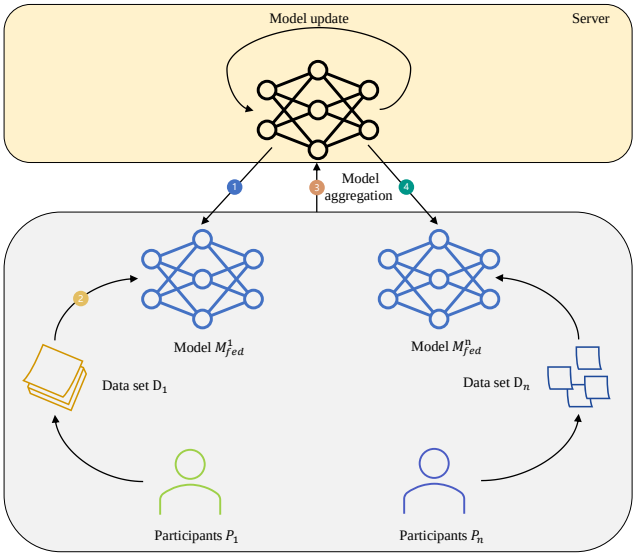


Fig. 1. The workflow of the federated learning. The illustration showcases four common steps in federated learning: Initial model download, Local gradient training and transmission, Model aggregation, and Federated model update.

The main process is divided into the following four steps:

1) Download the initial model: Multiple participants, denoted as P_i (with potentially different dataset sizes), receive a shared initial model M_{fed}^0 from the server as part of the federated learning process. This initial model serves as the starting point for the participants' training processes.
2) Train and send gradients locally: Each participant P_i, trains their local model using their respective dataset D_i. After training, they upload updated gradients to the server.
3) Model aggregation: The server collects the model gradients from each participant P_i, and aggregates them to generate a new federated model, denoted as M_{fed}^i. This aggregated model is then shared back with each participant P_i, ensuring that all participants have the updated model for the next iteration of training.
4) Update the federated model: This iterative process continues until the federated model converges, indicating that the training process has reached a satisfactory state. At this point, the training process is terminated, and the final federated model M_{fed} is obtained.

In this article, we focus on reviewing existing privacy inference attacks and defenses in both centralized learning and federated learning scenarios.

III. Privacy Attack Strategy

We have reviewed several important papers published in recent years that discuss privacy attacks in machine learning. These attacks can be classified into three categories: membership inference, model extraction, and property inference. We summarize these attacks in Table II. We have also divided machine learning into two main phases, namely the training and prediction phases. In the training phase, we discussed attacks in both the centralized learning environment and the federated learning environment. However, in the prediction phase, there is no such distinction. At this stage, privacy attacks mainly include two types: white-box and black-box attacks. In a white-box attack, the attacker knows the structure of the target model and can obtain its internal parameters. On the other hand, in a black-box attack, the attacker can only request the target classifier through the prediction API and obtain its corresponding confidence value.

A. Membership inference attack

The target of a membership inference attack is to determine whether or not a given simple exists in the training set. In Table III, we summarize the membership inference attack papers in recent years from task, approach, dataset, and classify them according to attack knowledge and scenarios. It is not difficult to see that most papers focus on the black-box model because attackers have less information than the white-box model, which means that these attacks are often more destructive. The characters "CL" and "FL" in Table III are short for centralized learning and federated learning, respectively.

1) Centralized Learning Scenario: Existing research [43, 44] has shown that machine learning models, such as Deep Neural Network (DNN), are often over-parameterized, meaning that they have more parameters than necessary for good performance. This can cause the model to memorize its training data and store it in the model parameters [16, 45]. Membership inference attacks can be classified into two categories based on different attack methods: shadow training-based attack approaches and metric-based attack approaches.

Shadow training Methods: Fig. 2 illustrates the steps involved in a membership inference attack using shadow training methods. Here, $D_1^{'}, ..., D_k^{'}$ represents a shadow training dataset that is distinct from the private training dataset, while $T_1, ..., T_k$ are shadow test sets that correspond to $D_1^{'}, ..., D_k^{'}$. The attacker trains each shadow model in a way that simulates the behavior of the target model. Once the shadow models have completed their training, the attacker queries each model using its respective shadow training and test datasets, obtaining an output. For the shadow training dataset, the prediction

TABLE II
SUMMARY OF INFERENCE ATTACK

Attacks	Key idea	Phase		Target information	
		Training	Prediction	Training data	Model
membership inference attack	determine whether or not a given simple exists in the training set		✓	✓	
model extraction attack	copy the functionality of the target model		✓		✓
property inference attack	infer confidential information about the training dataset that the model provider does not want to release	✓	✓	✓	

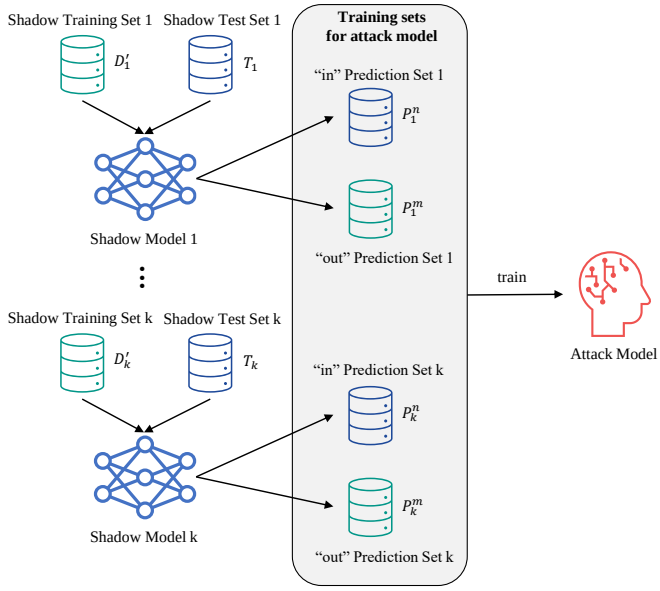


Fig. 2. Overview of the shadow training method

vector is labeled as "in", while for the shadow test dataset, the prediction vector is labeled as "out". The training dataset consists of k "in" and "out" data sets.

In 2017, Shokri et al. [16] first proposed a membership inference attack based on the black-box model. They use multiple shadow models to train a binary classifier as an attack model. Subsequently, Salem et al. [19] only require training a shadow model, without the need for knowledge about the model architecture and the distribution of training data. Later, Liu et al. [28] introduced EncoderMI, the first membership inference method for contrastive learning. In EncoderMI, a binary classifier is constructed by the attacker to predict whether the input belongs to the target encoder. In addition, membership inference attacks based on shadow models are also used in recommender systems [29] and graph neural networks [46].

Metric-based Methods: In addition to the shadow model training approach described above, there are now several papers focusing on the metric-based membership inference attacks. Membership inference attacks based on metrics involve computing a metric on the prediction vectors of a data record and comparing it to a predetermined threshold to determine its membership status. Such attacks are typically simpler and less computationally expensive than shadow model-based attacks. Song et al. [20] considered both the security and privacy domains together, and used model predictions to conduct an inference attack. They found that the adversarial defense method can increase the risk of membership inference attacks on the target model. Song et al. [32] argue that existing predictive entropy-based attacks do not take into account the ground truth labels, which can result in incorrect classification of members and non-members. To address this issue, they proposed a new inference attack based on modified prediction entropy that avoids assigning zero prediction entropy to both completely correct and completely wrong classifications. Yuan et al. [38] focused on pruning neural networks and proposed a self-attention membership inference attack. Their model makes a binary prediction of membership status by considering the prediction confidence and sensitivity of the pruned model along with the ground truth label.

Recently, several papers [30, 40, 47] have started to investigate a label-only model, in which attackers can only utilize labels for training and prediction purposes. Li et al. [30] develop two types of decision-based attacks, namely transfer attacks and boundary attacks. The transfer attack is still based on the shadow model, in which the shadow dataset is relabeled by the prediction label of the target model, transferring the membership information to the shadow model. The shadow model is then used to launch a score-based membership inference attack locally. The boundary attack involves perturbing candidate data samples to alter the prediction label of the model and using the magnitude of the perturbation to differentiate between member and non-member data samples. Zhang et al. [40] focused on the semantic segmentation and achieved this by extending the available data through data augmentation techniques. They employed various post-processing strategies to launch attacks on the target model.

Unlike the majority of membership inference attacks on classification models, Song et al. [24] focused on membership inference attacks on embedding models. They infer the membership of a context of data by using similarity scores of sliding windows of words or sentences. In their experiment, the adversary was able to achieve a 30% improvement in membership information over random guessing for both word and sentence embeddings. He et al. [25] and Shafran et al. [48] extend the membership inference attacks to the task of image segmentation. In [25], the authors explore attacks on segmentation using a shadow model and highlight the significance of spatial structures in segmentation attacks. In contrast, [48] does not require a large number of in-distribution data samples to train the shadow model. Instead, it operates on a single sample and uses only one query on the victim

TABLE III
SUMMARY OF MEMBERSHIP INFERENCE ATTACK

Ref.	Publication	Attack Knowledge		Scenarios		Task	Approach	Dataset
		Black-box	White-box	CL	FL			
[16]	2017, S&P	✓		✓		Classification	Shadow training	CIFAR, Purchases, Locations, Texas hospital stays, MNIST, UCI Adult
[17]	2018, NDSS		✓	✓		Classification	Distinguishing function	Transport For London, San Francisco Cabs
[18]	2019, ICMLC	✓		✓		Classification	Prediction loss	CIFAR-10,CIFAR-100
[19]	2019, NDSS	✓		✓		Classification	Prediction entropy Prediction confidence	Adult, News, MNIST, LFW CIFAR-10, CIFAR-100 Purchase-100, Foursquare
[20]	2019, CCS	✓		✓		Classification	Shadow training Prediction confidence	Fashion-MNIST Yale Face, CIFAR-10
[21]	2019, PoPETs	✓	✓	✓		Generation	Prediction confidence	EyePACS CIFAR-10, LFW
[22]	2019, S&P		✓	✓	✓	Classification	Intermediate computation	CIFAR-100, Purchase-100, Texas-100
[23]	2019, WCSP		✓		✓	Classification	Shadow training	MNIST, CelebA
[24]	2020, CCS	✓		✓		Embedding	Similarity score	Wikipedia, BookCorpus
[25]	2020, ECCV	✓		✓		Image segmentation	Shadow training	Mapillary-Vistas Cityscapes, BDD100K
[26]	2020, CCS	✓	✓	✓		Generation	Reconstruction error	CelebA, MIMIC-III Instagram New-York
[27]	2020, USENIX-Security		✓	✓		Classification	Idiosyncratic features	Adult Diabetes, LFW Cancer, Hepatitis CIFAR-10, CIFAR-100 MNIST, German credit
[28]	2021, CCS	✓		✓		Classification	Shadow training	CIFAR-10, STL10 Tiny-ImageNet
[29]	2021, CCS	✓		✓		Classification	Shadow training	Amazon Digital Music, Lastfm-2k, Movielens-1m
[30]	2021, CODASPY	✓		✓		Classification	Prediction confidence	CIFAR-10, CIFAR-100, GTSRB, Face
[31]	2021, IEEE TSC	✓		✓		Classification	Shadow training	Adult, CIFAR-10 MNIST, Purchase-100
[32]	2021, USENIX-Security	✓	✓	✓		Classification	Prediction entropy	CIFAR-100, Foursquare, Purchase-100, Texas-100
[33]	2022, CCS	✓		✓		Classification	Self distillation	Purchase-100, CIFAR-10, CIFAR-100, MNIST
[34]	2022, CCS	✓		✓		Classification	Loss trajectory	CIFAR-10, CINIC-10, CIFAR-100, GTSRB, Purchase, Location, News
[35]	2022, ICLR	✓		✓		Classification	Difficulty calibration	German Credit, Hepatitis, Adult, MNIST, CIFAR-10, CIFAR-100, ImageNet
[36]	2022, JISA	✓			✓	Classification	Prediction confidence	MNIST, CIFAR, Purchase-100, Texas-100
[37]	2022, S&P	✓		✓		Classification	Estimate the likelihood	CIFAR-10, CIFAR-100, ImageNet, WikiText-103
[38]	2022, USENIX-Security	✓		✓		Classification	Prediction confidence	CIFAR-10, CIFAR-100, CHMNIST, SVHN, Texas, Location, Purchase
[39]	2022, CCS	✓	✓	✓		Classification	Steal hyperparameters	CIFAR-10, CIFAR-100, TinyImageNet, Purchases, Locations, Texas
[40]	2023, TDSC	✓		✓		Semantic Segmentation	Shadow training	Cityscapes, BDD100K, Mapillary Vistas
[41]	2023, TDSC	✓		✓	✓	Classification	Learning logits distribution	Purchases-100, Texas-100, MNIST, CIFAR-10, CIFAR-100
[42]	2023, TIFS		✓		✓	Classification	Feature construction	CIFAR10, MNIST, Accident, Adult

model. Recently, researchers have extended membership inference attacks to several real-world scenarios. Niu et al. [49] generalized membership inference attacks to code models for extracting sensitive private information from these models. Furthermore, Shi et al. [50] utilized membership inference attacks to launch attacks on wireless signal classifiers.

2) Federated Learning Scenario: In 2019, Nasr et al. [22] were the first to propose white-box membership inference attacks under federated learning, using the gradient vector of the model over all parameters on the target data points as the primary features of the attack. As parameter updates in federated learning can affect other party parameters, they also devised an active attack to make stochastic gradient descent reveal even more information about the participant's data. Mao et al. [23] proposed a Generative Adversarial Network (GAN) model to generate a training dataset. The attacker deceives the victim into training the discriminator model of the GAN and uses the probability value output by the attack model as the prediction of the members of the training set. Gu et al. [36] proposed a membership inference attack based on a prediction confidence sequence, in which they utilized a fully connected network to process the sequence of prediction confidence and differentiated members from non-members based on the difference in the rate of increase of the prediction confidence of training data compared to that of test data. In Federated Learning scenario, attacks can be more easily implemented due to the gradual updating of the global model by each participant during the training process. On the other hand, in the centralized scenario, achieving white-box conditions for attacks is usually more difficult.

Comparisons and Summaries. We argue that one of the main reasons for membership inference attacks stems from the exploitation of the overfitting phenomenon inherent in the target model. Deep learning models, characterized by over-parameterization and heightened complexity, exhibit a robust capability to memorize intricate details or noise within a given dataset [51]. Moreover, the iterative nature of machine learning model training, involving repeated exposure to the same samples over numerous epochs, leads to the model easily retaining information about these training samples. Consequently, attackers find it advantageous to discern members from non-members. In federated learning scenario, where data is decentralized among multiple participants, membership inference encounters an increased level of complexity, presenting a formidable challenge for attackers.

B. Property inference attack

Property inference attacks aim to infer confidential information about the training dataset that the model provider does not want to release, such as age or gender distribution. These attacks can occur during the model training phase or after the model has been trained. The latter is typically referred to as a model inversion attack. In Table IV, we summarize recent research on property inference attacks, and compare and summarize these studies based on various criteria such as Attack Knowledge, Scenarios, Target Information, Approach, and Dataset.

1) Centralized Learning Scenario: **property inference attack:** Yeom et al. [53] focused on the impact of overfitting on machine learning privacy. Their formal analysis showed that attribute reasoning is highly sensitive to overfitting, and models become more vulnerable to attack as they overfit more. On this basis, Jayaraman et al. [57] investigated a fine-grained variant of attribute reasoning called sensitive value reasoning. In this type of attack, the goal of the attacker is to identify records from a candidate set with high confidence, where a specific sensitive value is associated with an unknown attribute. Compared to attribute reasoning attacks, sensitive value reasoning is more targeted and focused on specific values of a sensitive attribute. Their white-box attack is not affected by the presence or absence of a single candidate record in the training dataset, which suggests that the attack can reveal the underlying training distribution of the model rather than knowledge of a specific training record.

Recently, Wang et al. [58] focused on group property inference attacks on graph neural networks, which aim to infer the distribution of specific node and link groups in the training graph, such as links between male nodes being more links than those between female nodes. They found three reasons why the group property inference attack is successful: Firstly, the correlation between the property feature and the label. Secondly, the non-negligible disparity in the influence of different node/link groups on the target model. And thirdly, the model parameters and outputs of the target model trained on the data with the target property $\mathbb{P}$ are distinctly dissimilar to those obtained from the data without $\mathbb{P}$. They verified the effectiveness of these attacks in both black-box and white-box settings, achieving promising results.

Moreover, Zhou et al. [59] have raised concerns about the security and privacy risks associated with GANs. In their study, they proposed property inference attacks against GANs, whereby the attacker first queries the target GAN model to obtain a set of generated samples. Subsequently, the attacker employs a property classifier to label these samples concerning the target property. Finally, the attacker infers the target property by summarizing the results of the property classifier. They have extensively evaluated their model on various tasks using real-world datasets, such as MNIST, CelebA, AFAD, and US Census Income, in both full black-box and partial black-box settings, demonstrating excellent attack performance.

model inversion attack: Machine learning is becoming a commodity with many users relying on online machine learning applications to simplify their training operations. However, this also brings new privacy risks. Song et al. [52] hypothesized a scenario where a malicious machine learning provider may perform black or white-box attacks on the resulting model. Under a white-box attack, the attacker can access the entire model. Song et al. [52] proposed three attacks: 1) encoding sensitive information of the training dataset directly in the least significant bits of the model parameters, 2) making the parameters highly correlated with the sensitive information, and 3) encoding the sensitive information in the symbols of the parameters. Under a black-box attack, the attacker can only access the predicted interface. In this case, they proposed a technique similar to data augmentation.

TABLE IV
SUMMARY OF PROPERTY INFERENCE ATTACK

Ref.	Publication	Attack Knowledge	Scenarios	Target information	Approach	Dataset
[52]	2017, CCS	Black-box White-box	CL	Model's training data	LSB encoding; Correlated value encoding; Sign encoding	CIFAR-10, LFW, FaceScrub, 20 Newsgroups, IMDB Movie Reviews
[53]	2018, CSF	Black-box	CL	Victim participant's training data	Using Model Overfitting	Eyedata, IWPC, Netflix, MNIST, CIFAR-10, CIFAR-100
[54]	2018, CCS	White-box	CL	Model's training data	Neuron sorting; Set-based representation	US Census Income, MNIST, CelebA, HPCs
[55]	2019, arXiv	Black-box	CL	Model's training data	Training-based strategy values	FaceScurb, CelebA, CIFAR-10, MNIST
[56]	2020, USENIX Security	Black-box	CL	Model's training data	Conditional Best of Many GAN	MNIST, CIFAR-10, Insta-NY
[57]	2022, CCS	White-box	CL	Records with sensitive attributes	Recognition related neuron	Texas-100X, Census19
[58]	2022, CCS	Black-box White-box	CL	Particular nodes distribution	Training attack classifier	Pokec, Facebook, Pubmed
[59]	2022, NDSS	Black-box	CL	Model's training data	Training Shadow GAN	MNIST, CelebA, AFAD, US Census Income
[60]	2022, S&P	Black-box	CL	Model's training data	Poisoning attack	Census, Enron
[61]	2022, USENIX Security	Black-box	CL	Records with sensitive attributes	Analyze the confidence score	General Social Survey, Adult, Fivethirtyeight
[62]	2017, CCS	White-box	FL	Victim participant's training data	GAN	MNIST, AT&T dataset of faces
[14]	2019, S&P	White-box	FL	Victim participant's training data	Training a meta-classifier	LFW, PIPA, FaceScrub, CSI,FourSquare, Yelp-health, Yelp-author
[63]	2019, INFOCOM	Black-box	FL	Model's training data	Design a multi-task GAN	MNIST, AT&T
[64]	2019, ACSAC	Black-box White-box	FL	Model's training data	Regularized Maximum Likelihood Estimation; Inverse-Network	MNIST, CIFAR-10
[65]	2019, NeurIPS	Black-box	FL	Model's training data	Deep Leakage from Gradients	MNIST, CIFAR-100, SVHN, LFW
[66]	2020, arXiv	Black-box	FL	Sample's label	Matching virtual and shared gradients	MNIST, CIFAR-100, LFW
[67]	2020, NeurIPS	Black-box	FL	Model's training data	Numerical reconstruction	CIFAR-10
[68]	2021, ICDE	Black-box	FL	Sample's label	Equality solving; Path Restriction	bank marketing, credit card,drive diagnosis, news popularity
[69]	2022, USENIX security	White-box	FL	Sample's label	Direct/passive/active label inference attack	CIFAR-10, CIFAR-100, CINIC-10,BHI Yahoo Answers, Criteo

Large-scale data has become increasingly important in recent machine learning, and data collection is a continuous process, which means that machine learning model owners frequently update their models with newly collected data. Salem et al. [56] focused on whether changes in the output of the black-box machine learning model before and after an update would reveal information about the dataset used to perform the update, namely the updating set. They proposed four different attacks in this context, which can be categorized into two classes: single-sample attack class and multi-sample attack class. The two attacks in the single-sample attack class focus on the simplified case when the target machine learning model is updated with a single data sample, while the two attacks in the multi-sample attack class deal with the more general and complex cases when the update set contains multiple data samples. These four attacks are single-sample label inference attack, single-sample reconstruction attack, multi-sample label distribution estimation attack, and multi-sample reconstruction attack. To perform these attacks, Salem et al. [56] randomly select a fixed set of data samples, referred to as the "probing set", and evaluate two different versions of the target machine learning model, the initial model, and its updated version. They use a Multi-Layer Perceptron to implement an encoder that takes as input the difference in the output of the two

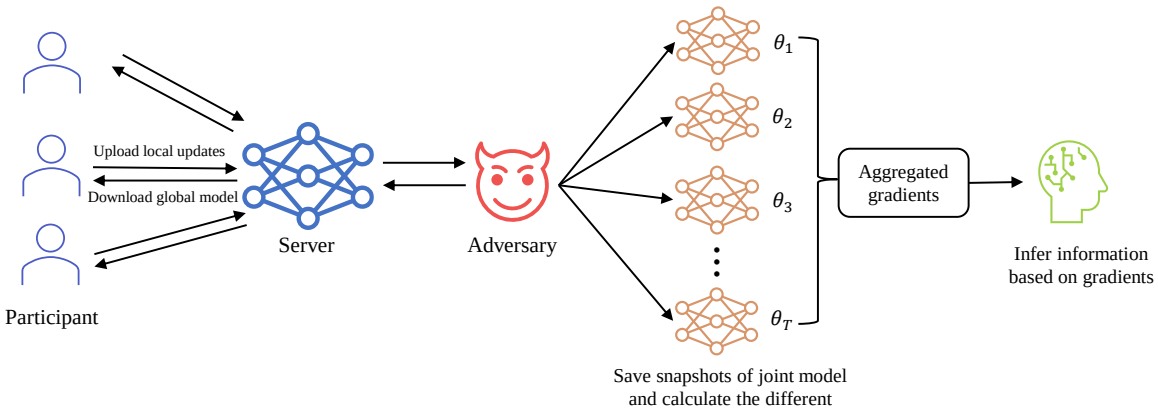


Fig. 3. Overview of the property inference attack

versions of the model, namely the posterior difference. Finally, the decoder generates various types of information about the update set for each attack. The quantitative and qualitative results demonstrate that these attacks perform well.

Recently, Mehnaz et al. [61] proposed the Confidence Score Based Model Inversion Attack (CSMIA) and the Label Only Model Inversion Attack (LOMIA) as novel attacks. Notably, they are the first to propose work on label-only Model Inversion Attack and CSMIA attacks assume that the adversary can access the confidence scores of the target model, whereas LOMIA attacks assume that the adversary can only access the label predictions of the target model. The researchers discovered that some subsets of the training data, grouped by attributes such as gender or race, may be more susceptible to model inversion attacks than others, which they termed "disparate vulnerability". They further conducted extensive evaluations of the attack on decision trees and deep neural networks. The experimental results demonstrated that their attack outperformed existing attacks in both binary and multivalued sensitive attribute inference.

2) Federated Learning Scenario: As the primary difference between federated learning and centralized learning lies in the training process, there is no distinction between property inference attacks and model inversion attacks in the reasoning process. Thus, we do not distinguish between these two attack types in this section. Fig. 3 illustrates the fundamental process of a property inference attack in federated learning. In each iteration, the participant downloads the latest federated model, calculates the gradient updates, and transmits them to the server. The attacker, in addition to these actions, also saves a snapshot of the federated model parameters. The attacker exploits the combined updates of all participants to launch property inference attacks.

In 2019, Melis et al. [14] conducted a study on both passive and active property inference attacks in federated learning. They discovered that deep learning models have independent internal representations for different types of features, some of which may not be relevant to the learning task. These "unintended" features can be exploited by attackers to reveal information about participants' training data. In a passive attack, an attacker takes a snapshot of the global model and generates aggregate updates to differentiate between data with and without the attribute of interest. This produces labeled examples, allowing the attacker to train a binary property classifier that can determine whether an observed update is based on data with or without that attribute. In an active attack, the only difference is that the attacker performs additional local computations and submits the resulting values into the federated learning protocol, while a passive attack does not alter anything in the local or global collaborative training process.

In addition to malicious client-generated attacks, it is also crucial to address the issue of malicious servers. Wang et al. [63] specifically investigated property attacks initiated by malicious servers in the context of federated learning. They introduce a framework called multi-task GAN for Auxiliary Identification (mGAN-AI), which combines GAN with a multi-task discriminator capable of simultaneously discriminating the category, reality, and client identity of input samples. Using mGAN-AI on the MNIST and AT&T datasets, they successfully recover user-specific samples.

In recent studies, significant attention has been given to exploring the privacy risks associated with horizontal federated learning, while the privacy risks of vertical federated learning (VFL) have received less comprehensive examination. Specifically, Luo et al. [68] were the pioneers in proposing a property inference attack tailored for VFL. For logistic regression (LR) models, they discussed equation-solving attacks. For the decision tree model, they proposed a path restriction attack. Additionally, for more complex models such as neural networks (NN) and random forests (RF), they developed a general-purpose feature inference attack based on multi-model prediction, known as the generative regression network attack. Subsequently, Fu et al. [69] have introduced a label inference attack against VFL. They came up with three effective attacks. Firstly, they proposed the Passive Label Inference Attack through Model Completion, which uses a small set of labeled data to add a classification layer to the local model for label inference. Secondly, they developed the Active Label Inference Attack with the Malicious Local Optimizer, which designed an adaptive malicious local optimizer to amplify the gradient of each parameter in the opponent's bottom model. The opponent

can train a bottom model with more label hidden information by using the malicious local optimizer in the training phase. Then, in a passive inference attack, the adversary can complete the model to obtain the final tag inference model. Finally, they proposed the Direct Label Inference Attack, which infers the corresponding label based on the gradient received from the prediction layer.

In the context of federated learning, the technique of gradient exchange has been subject to inference attacks, as demonstrated by the works of Zhu et al. [65] and Zhao et al. [66]. Zhu et al. [65] performed the standard forward and backward passes using a randomly generated pair of "dummy" inputs and labels. Virtual gradients were derived from the virtual data, and virtual inputs and labels were optimized to minimize the distance between the virtual and real gradients. By matching the gradients, the virtual data was brought closer to the original data. Zhao et al. [66] were able to extract the true labels entirely by uncovering the correlation between the labels and gradient symbols, building on the original approach.

Comparisons and Summaries. Property inference attacks aim to infer irrelevant attributes rather than common features contributing to the target's learning. In this context, attackers amass extensive training datasets and subsequently employ them to locally train meta-classifiers. These meta-classifiers are designed to ascertain whether the target classifiers exhibit specific attributes. Much like the conundrum encountered in membership inference attacks, property inference attacks grapple with the complexities of a federated learning scenario where data is distributed across multiple participants.

C. Model extraction attack

As the cost of training machine learning models increases and machine learning as a service (MLaaS) becomes more popular, machine learning models have become valuable assets. However, this has also led to a new privacy risk as attackers can use model extraction attacks to copy the functionality of the target model, essentially stealing it. Typically, this type of attack occurs during the inference phase, after the model has already been trained. Therefore, in this section, we do not distinguish between centralized learning and federated learning scenarios. In Table V, we summarize recent research on model extraction attacks and compare and summarize these studies based on various criteria such as Attack Knowledge, Stealing Information, Approach, and Dataset. It is important to note that some studies fall under the category of Gray-box attacks, which involve having some knowledge about the victim model, while Black-box attacks involve having no information about the victim model.

In 2018, Wang et al. [70] emphasized the importance of hyperparameters in machine learning algorithms, as different hyperparameters can lead to significantly different model performances on the same training dataset. Hence, hyperparameters can be considered confidential information. To extract this information, they utilized the fact that the gradient of the objective function is approximately zero at the optimal model parameters, and derived a system of linear equations to determine the hyperparameters. They then applied the linear least squares method[82] to solve for the hyperparameters.

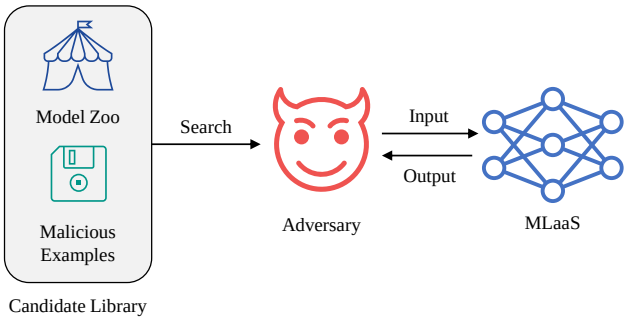


Fig. 4. Illustration of the model extraction attack through adversarial examples.

In 2019, Orekondy et al. [71] researched stealing model functionality, focusing on an attack method under a black box model. They first queried the black-box model for a set of input images to obtain predictions and then trained a "knockoff" model with the queried pairs of image predictions. Due to the effectiveness of the counterfeit model, they were able to query the victim model for only $30, highlighting the potential threat of model extraction.

Later, Yu et al. [72] conducted a study to investigate the extent to which model information can be leaked by public prediction APIs under black box models in the real world. They carried out model extraction attacks on five commercially hosted MLaaS platforms, which included Microsoft, IBM, and Google, and demonstrated the effectiveness of their attacks. The researchers found that existing model extraction attacks typically target small-scale machine learning models, and require a large number of queries against the target model, which can become prohibitively expensive for models with millions of parameters[83]. To address this, they proposed a "Knockoff"-like[71] attack, as shown in Fig. 4, which uses input-output pairs to retrain the surrogate model. They also applied a margin-based adversarial active learning algorithm to search for the model zoo and malicious examples and queried the victim model using these samples to effectively estimate the distance between the decision boundary of the victim model and the stolen model, thereby reducing the number of queries required and accelerating the stealing speed. [73] and [75] also used these techniques.

Moreover, Rakin et al. [80] focused on hardware-based DNN attacks and found that adversaries can use side-channel attacks to obtain sensitive information in target systems. Unlike most existing attacks that focus on inferring high-level model information such as model architectures, they focused on inferring detailed model parameters such as weights and biases. They developed HammerLeak, which exploits fault-based information leakage to effectively steal partial model weight parameters on a large scale through Rowhammer. Subsequently, they obtained a high-fidelity victim surrogate model using a surrogate model training algorithm based on mean clustering weight penalty. The surrogate model successfully achieved a test accuracy of over 90% on the deep residual network of the CIFAR-10 dataset.

In addition to their research on model extraction attacks in image and text classification, Yue et al. [77] also studied

TABLE V
SUMMARY OF MODEL EXTRACTION ATTACK

Ref.	Publication	Attack Knowledge	Stealing information	Approach	Dataset
[70]	2018, S&P	Gray-box	Hyperparameters	Linear least square approach	Diabetes, GeoOrig, Bank UJIIndoor,Iris, Madelon
[71]	2019, CVPR	Black-box	Functionality	Training "Knockoff" classifier	Caltech256, CUBS200, Indoor67, Diabetic5
[72]	2020, NDSS	Black-box	Functionality	Malicious samples query	GTSRB, VGG Flowers, KDEF
[73]	2020, AAAI	Black-box	Functionality	Build universal thief datasets	MNIST, CIFAR-10, GTSRB, MR, IMDB, AG News
[74]	2020, USENIX security	White-box	Functionality	Query synthesis active learning	scikit-learn; UCI machine learning repository
[75]	2020, EDSMLS	Black-box	Functionality	Given more realistic assumptions	Caltech, CUBS Diabetic5, GTSRB, CIFAR 10
[76]	2020, USENIX security	Black-box	Weight recovery; Functionality	Learning-based strategy; Extraction model's weights	MINST, CIFAR-10
[77]	2021, RecSys	Black-box	Weight recovery	Autoregressive generation	Movielens-1M, Steam, Amazon Beauty
[78]	2021, ICML	Gray-box	Functionality	Fine-tuned encoder; Algebraic attack	SST-2, MNLI
[79]	2022, ICML	Black-box	Encoders; Supervised Models	Direct Extraction; Recreate Projection Head	ImageNet, CIFAR10, SVHN
[80]	2022, S&P	Gray-box	Weight recovery	Fault-based information leakage	CIFAR-10, CIFAR-10, GTSRB
[81]	2022, IEEE Trans. Fuzzy Syst.	Black-box	Structure;Weight; Metaparameter	Fuzzy gray correlation	Hand-written digit dataset in Scikit-learn

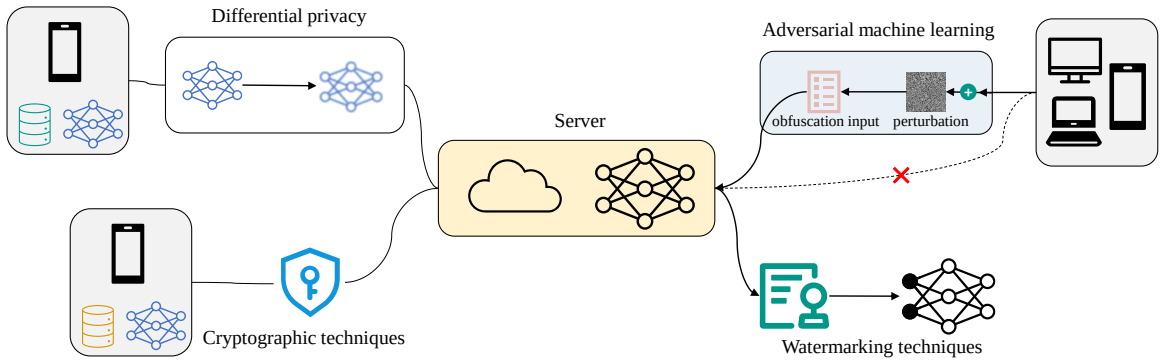


Fig. 5. An overview of the privacy defense. We summarise four common defence methods in the figure, including Differential privacy, Cryptographic techniques, Adversarial machine learning and Watermarking techniques.

model extraction attacks in sequential recommender systems. They addressed the challenge of applying model extraction attacks on sequential recommendation systems, which cannot use proxy datasets with semantic similarities like in image or text tasks. Instead, they utilized autoregressive generation to synthesize data and employed knowledge distillation to quickly narrow the gap between the victim and attacker recommenders, since APIs typically only provide rankings, making it difficult to attack sequential recommendation systems.

Chandrasekaran et al. [74] proposed a model extraction attack using active learning, which allows the attacker to train attack models on unlabeled data using an active learning approach. This method is classified as a white-box model, as it assumes that the attacker can generate arbitrary query instances. Jagielski et al. [76] developed an improved training-based model extraction attack that directly extracts model weights without the need for training, resulting in increased attack efficiency.

Comparisons and Summaries. With the escalating volume of training data, the associated training time also surges, underscoring the growing significance of machine learning models. Model extraction attacks serve as a means for attackers to acquire the model without incurring these training costs. Moreover, through model extraction, attackers gain access to

TABLE VI
AN OVERVIEW OF THE PRIVACY DEFENSE

Defenses	Key idea	Phase		Target information	
		Training	Prediction	Training data	Model
Differential privacy	adding random noise to hide or blur the actual results	✓		✓	
Cryptographic techniques	encrypt data using cryptographic knowledge to avoid privacy breaches	✓	✓	✓	✓
Adversarial machine learning	defending against privacy attacks by adopting an adversary's perspective.	✓		✓	
Watermarking techniques	add unique identifiers to models to prevent model leakage		✓		✓

the model's private data or can circumvent the model's security policies.

D. Other inference attack

Several new inference attacks have been recently proposed. In 2021, Hu et al. [84] introduced the Source Inference Attack (SIA) in federated learning, which aims to identify the origin of training members. They demonstrated that a central server, honest but curious, could perform an effective SIA non-intrusively by estimating the source of a training member optimally, using the prediction loss of local models. In 2022, Gadotti et al. [85] proposed the Pool Inference Attack, which quantifies the sensitive information leaked by local differential privacy mechanisms in practice. The adversary infers the user's preferred pool, which refers to disjoint groups of objects, by identifying which pool the user is most likely to choose.

IV. PRIVACY DEFENSE STRATEGY

Due to the increasing number of privacy attacks targeting sensitive information, there has been a significant focus on developing privacy-preserving machine learning algorithms. These defense methods can be categorized into four main strategies: differential privacy, cryptographic techniques, adversarial machine learning, and other defense techniques. These strategies are applied at different stages of the machine-learning process to safeguard users' private information and prevent data leakage. Fig. 5 summarizes the prominent types of defensive mechanisms in Machine learning.

A. Differential privacy

Differential privacy has been a commonly used tool for preserving the privacy of deep learning models[99]. It involves adding random noise to hide or blur the actual results of query operations until attackers are unable to distinguish them, thus protecting sensitive data. Generally, noise can be added to shared gradients in federated learning to protect participants' privacy[95, 98]. Additionally, noise can be applied to datasets, loss functions, and model weights. Differential privacy can be used to defend against attacks such as model extraction and membership inference attacks. As shown in Table VII, we summarize several privacy-preserving machine learning methods based on the differential privacy in terms of experimental datasets, models, target-protected information, and approaches.

Centralized Learning Scenario: In 2019, Phan et al. [87] introduced a novel approach called the heterogeneous Gaussian mechanism to enhance the privacy of deep neural networks. This technique aims to strike a balance between model utility and the potential loss of privacy by redistributing noise across the first hidden layer and the model gradient in a flexible manner. Specifically, they applied Gaussian noise injection to the first hidden layer, which served to enhance the robustness of the model against attacks. By carefully managing the distribution of noise, the authors were able to achieve both effective privacy protection and satisfactory model performance.

Later, Cao et al. [89] shifted their focus towards the privacy protection of training data. They proposed a method that minimizes the Sinkhorn divergence, an efficient approximation of the optimal transport distance, between the model and data while preserving differential privacy. They also introduced a novel technique to manage the bias-variance trade-off of gradient estimates. By adopting these approaches, the training data is obfuscated to prevent potential privacy breaches and maintain data confidentiality during the learning process.

Moreover, Yan et al. [90] address the challenge of defending against model extraction attacks. They identify the limitations of traditional differential privacy mechanisms, which are either fragile or require a large privacy budget when facing common model extraction and multiple query attacks. In response, they propose a novel real-time model extraction state evaluation mechanism called Monitor. This mechanism assesses the state of model extraction. Based on the Monitor's results, they employ an adaptive differential privacy budget allocation method to dynamically adjust the amount of noise added to the model's responses. This approach aims to maximize the model's usability while effectively preserving its privacy. By dynamically allocating the privacy budget, the model can maintain a balance between utility and privacy, offering robust protection against model extraction attacks.

Similar to Yan et al. [90], Ye et al. [91] also explore the significance of privacy budgets in defending against privacy attacks. Their approach involves obfuscating the attacker's classifier through modifications and normalization of the confidence score vector. They ensure that the order of scores in the vector is preserved, thereby guaranteeing zero classification accuracy loss. By carefully manipulating the confidence scores while maintaining their relative order, the attacker's ability to extract sensitive information from the classifier is hindered, effectively protecting the privacy of the model.

TABLE VII
SUMMARY OF DIFFERENTIAL PRIVACY

Ref.	Publication	Scenarios	Target protected information	Model	Approach	Dataset
[86]	2016, CCS	CL	Gradients	NN, CNN	Differentially private SGD	MNIST, CIFAR-10
[87]	2019, IJCAI	CL	Weights of neural networks	CNN	Heterogeneous Gaussian mechanism	MNIST, CIFAR-10
[88]	2020, NeurIPS	CL FL	Part of the gradients	MLP, CNN, 11 scikit-learn classifiers	Selectively applying DP SGD	MNIST, Fashion-MNIST
[89]	2021, NeurIPS	CL	Clients' datasets	Logistic regression, MLP, CNN	Based on optimal transport	MNIST, Fashion-MNIST, CelebA
[90]	2022, TDSC	CL	Model	LR, NN	adjust the privacy budget dynamically	SocialAds, Titanic, Email Spam, Mushrooms
[91]	2022, TIFS	CL	Clients' parameters	CNN	Change confidence score vectors	MNIST, Fashion-MNIST, CIFAR10
[92]	2015, CCS	FL	Gradients	MLP, CNN	Distributed selective SGD	MNIST, SVHN
[93]	2017, NeurIPS	FL	Clients' datasets	Linear regression	combines SMC with DP Bayesian learning methods	Wine Quality, Abalone data sets, GDSC
[94]	2018, NeurIPS	FL	Clients' contributions	CNN	Randomized mechanism	MNIST
[95]	2018, NeurIPS	FL	Output, Gradient	Logistic and linear regression	Output Perturbation, Gradient Perturbation	KDDCup9, KDDCup98
[96]	2020, TIFS	FL	Gradients	MLP, CNN	perturb the objective function of the neural network	ntegrated Public Use Microdata Series, MNIST, SVHN
[97]	2020, TIFS	FL	Clients' parameters	MLP	K-random scheduling	MNIST
[98]	2022, The VLDB Journal	FL	Gradients	GNN, DNN	Random adding noise	Frappe, Movielens

Federated Learning Scenario: Heikkila et al. [93] introduced a comprehensive approach for preserving privacy in learning patterns within distributed environments. Their method relies on a learning strategy designed for secure multi-party sum functions and integrates various private Bayesian learning techniques. In their approach, each client adds Gaussian noise to their data and partitions the noisy data into multiple segments using the fixed-point representation of real numbers. This process effectively cancels out the noise present in each segment. Subsequently, the individual shares are transmitted to a server. When the shares are combined, the sum accurately reveals the true value, while each share taken separately appears as random noise, ensuring privacy is preserved.

Later, Geyer et al. [94] aimed to protect client participation privacy in federated learning to ensure that the learning model does not disclose whether a client has participated. To achieve this, they incorporated differential privacy into federated learning by introducing Random Sub-sampling and Distorting techniques to alter and approximate the federated learning gradient aggregation. By employing these methods, their proposed approach can maintain differential privacy at the individual client level, with only a minimal impact on the model's performance, given a sufficient number of participating clients.

Moreover, Zhao et al. [96] presented a collaborative deep learning system with a focus on privacy preservation. To safeguard the sensitive information contained in the parameters, their system adopts a collective learning model, where only the parameters are shared, and not the actual data. Additionally, they propose the utilization of a functional mechanism to perturb the objective function of the network. By training the model on the perturbed objective function, they aim to enhance privacy protection while still achieving effective learning outcomes.

Recently, Wang et al. [98] introduced DP-PrivRec as a solution to address the issue of malicious actors in federated learning. Their approach focuses on preserving privacy at the user level, rather than individual data points. They achieve this by applying a Gaussian mechanism to add noise to the gradients transmitted from federated learning clients to the central server. In order to mitigate the performance degradation caused by privacy protection, they enhance item representation learning using a two-stage federated learning training method. This allows them to strike a balance between privacy preservation and maintaining the effectiveness of the federated learning model.

Comparisons and Summaries. Differential privacy is a proactive defense method that is actively introduced by the model owner during the training process. Its purpose is to protect against attacks by adding perturbations. Typically, the defender introduces perturbations at three different levels: gradients, datasets, and model parameter weights. However, it's important to note that perturbation can lead to performance degradation, which requires the defender to find a balance between model performance and privacy.

TABLE VIII
SUMMARY OF CRYPTOGRAPHIC TECHNIQUES

Ref.	Publication	Scenarios	Model	Approach	Dataset
[100]	2017, CCS	FL	Generalized	OT; 2PC	MNIST, CIFAR-10, PTB
[101]	2017, S&P	FL	NN	3PC	MNIST, Gisette
[102]	2017, CCS	FL	Generalized	SS; Blinding	-
[103]	2018, CCS	FL	NN, CNN	SS; 3PC	Synthetic datasets, MNIST
[104]	2019, CCS	FL	DNN	2PC	MNIST, MotionSense, Thyroid, Breast cancer, Skin Cancer MNIST, German credit
[105]	2020, CCS	FL	DNN	2PC	SqueezeNet, ResNet50, DenseNet121
[106]	2020, S&P	FL	Generalized	SMC	-
[107]	2022, USENIX Security	FL	NN	2PC	-
[108]	2023, TIFS	FL	NN	FSS; 2PC	Iris, Heart Disease, Bank Marketing, Credit Card, Handwritten Digits
[109]	2023, TIFS	FL	NN	FSS	MNIST, CIFAR-10, CIFAR-100, ImageNet
[110]	2018, CCS	CL	CNN	HE	MINST
[111]	2019, CCS	CL	CNN	MKHE	MNIST
[112]	2020, USENIX Security	FL	CNN	HE	FMNIST, CIFAR, LSTM
[113]	2021, MICRO	CL	Generalized	FHE	-
[114]	2022, CCS	CL	DT	FHE	UCI repository
[115]	2023, TIFS	CL	CNN	FHE	CIFAR-10, CIFAR-100

B. Cryptographic techniques

Cryptography plays a significant role in machine learning by providing various techniques to ensure secure and efficient operations without compromising sensitive information or sacrificing functionality. As listed in Table VIII, in this section, we focus on two common cryptographic techniques: secure multi-party computation(SMC) and homomorphic encryption(HE). By incorporating these cryptographic techniques into machine learning, users can benefit from secure, efficient, and accurate learning processes without compromising sensitive information or sacrificing functionality and efficiency.

1) Secure multi-party computation: SMC enables multiple parties to collaboratively compute a function over their private inputs without revealing individual inputs. This technique allows parties to perform joint computations while preserving the confidentiality of their data. It ensures that no party can learn anything beyond the final result of the computation.

Agrawal et al. [104] recognized the existing focus of security protocols for machine learning tasks mainly targeting the prediction level. To address this limitation, they proposed a novel protocol that optimizes deep neural networks at the training level. Their custom secure two-party protocol, QUOTIENT, was designed to enable secure computations during training. They also developed a semi-honest secure computation implementation called 2PC-QUOTIENT. To enhance efficiency, they combined Boolean sharing and additive sharing techniques, which facilitated the design of a dedicated protocol for efficient ternary matrix-vector multiplication based on correlation forgetting transfer. Additionally, they introduced efficient substitutions for quantization and normalization operations commonly used in machine learning, aiming to minimize computational overhead while preserving the security guarantees of the protocol. Finally, they devised a new fixed-point optimization algorithm to further improve the efficiency of the QUOTIENT protocol. The feasibility and effectiveness of QUOTIENT were demonstrated through extensive testing on real-world datasets.

However, the performance of 2PC-NN still falls short in terms of efficiency. For instance, when using CrypTFlow2[105], it can take more than 15 minutes for the server and client to execute and exchange over 30 GB of messages in order to perform a secure inference once on ResNet50. In order to address this challenge, Huang et al. [107] introduced a secure and high-speed two-party reasoning system called Cheetah1. To mitigate the need for multiple rotation operations arising from the spatial properties of convolution and matrix-vector multiplication, they employed three pairs of encoding functions to eliminate the need for rotation. Furthermore, they optimized the model's performance by employing a reduction protocol for nonlinear functions.

Likewise, Chen et al. [108] focus on enhancing the speed of 2-PC. They devised PriVDT, a streamlined two-party cryptographic framework tailored for training and reasoning

with vertical Decision Trees. To achieve this, they leveraged the novel Function Secret Sharing (FSS) protocol, utilizing comparison and division as fundamental components for optimal segmentation selection. Additionally, they introduced an efficient and privacy-enhancing partitioning protocol based on the iterative Goldschmidt paradigm. By decomposing the divisor into substrings, they evaluated a range of values with smaller bit lengths while concealing intermediate values, thereby improving the operational efficiency of the model.

2) Homomorphic encryption: HE is a powerful technique that allows computations to be performed directly on encrypted data. It enables computations on encrypted data without decrypting it, preserving privacy throughout the computation process. With homomorphic encryption, machine learning algorithms can be applied to encrypted data, ensuring privacy while still achieving accurate results. HE can also be seamlessly applied to federated learning. Prior to commencing training, the HE key pair is securely shared among all clients through a protected communication channel. Throughout the training process, each client encrypts its gradient updates using the public key and transmits the resulting ciphertext to the central server. The server aggregates the encrypted gradients received from all clients and disseminates the consolidated results back to each respective client. Upon receipt, the client decrypts the aggregated gradient using the private key, updates its local model accordingly, and proceeds to the subsequent iteration. Notably, as the client exclusively uploads encrypted updates, no discernible information is exposed to the server or any external entities during the data transfer and aggregation processes.

However, this approach often leads to a significant portion of time being consumed by encryption and decryption operations. Moreover, the encryption process generates much larger ciphertext, resulting in a substantially larger data volume compared to plaintext transmission. In order to address these challenges, Zhang et al. [112] introduced an innovative system solution called BatchCrypt for cross-context federated learning. In BatchCrypt, a batch of gradients is encoded as a long integer and encrypted only once, eliminating the need for full-precision encryption of individual gradients. Furthermore, the researchers developed novel techniques for quantization, coding, and gradient clipping, which effectively reduce the encryption and communication overhead associated with HE. As a result, the overall efficiency of the encryption process is greatly enhanced.

Kim et al. [115] also focused on enhancing the computational efficiency of FHE, specifically in the context of CNN inference. Their approach involved packing each input into the coefficients of a ring polynomial, enabling the evaluation of a single convolution through a single multiplication operation without any rotation. This technique resulted in a compact representation of the convolution operation. Additionally, they employed batch convolution, maximizing the number of convolution outputs packed into each output ciphertext. To further optimize the process, they made modifications to the bootstrapping process of FHE, enabling the evaluation of convolution and activation functions on different domains. As a result of these advancements, they achieved an efficient FHE scheme with a constant computational cost, independent of the kernel size. Notably, their approach demonstrated CNN inference speeds on CIFAR 10/100 that were at least five times faster than previous methods while maintaining comparable or higher accuracy levels.

Recently, Cong et al. [114] directed their attention toward decision tree algorithms and introduced a non-interactive design of private decision tree evaluation (PDTE) using FHE technology, named SortingHat. Their approach aimed at improving efficiency in this context. Firstly, they proposed a novel homomorphic comparison algorithm that enables comparisons between a ciphertext and a plaintext, effectively reducing the number of required polynomial multiplications. Furthermore, they developed an efficient technique for evaluating binary decision trees, referred to as homomorphic traversal, which only necessitates one outer product per decision node. Lastly, they enhanced the communication cost and time complexity of transmission by applying homomorphic comparison to the FiLIP stream cipher. These advancements collectively contributed to the improved efficiency and effectiveness of PDTE based on FHE technology.

Comparisons and Summaries. By incorporating cryptographic techniques into machine learning, we can achieve considerable privacy protection. However, it is important to strike a balance between encryption level and performance overhead. Setting encryption levels too high can result in significant performance degradation. This is particularly relevant in federated learning scenarios where gradients from all participants need to be encrypted. Moreover, excessive encryption can lead to larger ciphertext sizes, making decryption more challenging.

C. Adversarial machine learning

Adversarial machine learning is a field that focuses on defending against privacy attacks by adopting an adversary's perspective. By considering the defender as an attacker, it allows for a deeper understanding of the strategies and techniques employed by malicious attackers. This approach enables the defender to gain insights into potential vulnerabilities and threats to privacy, thereby enhancing their ability to implement robust defense mechanisms. By adopting an adversarial mindset, defenders can proactively strengthen privacy protection measures and effectively safeguard sensitive information. In Table IX, we assess recent papers on adversarial machine learning based on the following criteria: Target protected information, Model, Approach, and Dataset.

In recent years, the popularity of MLaaS has grown significantly. However, when users utilize machine learning services, uploading their data to the cloud can pose privacy risks. Without adequate protection, attackers may directly access raw data and uncover private information. In the past, researchers have proposed various techniques to obfuscate data in order to mitigate these risks. These techniques include homomorphic encryption[110], deletion of private areas[123], and uploading only necessary information for the model[124]. However, a study[125] has shown that even obfuscated data can still be vulnerable to privacy breaches. To address this challenge, Lin et al. [121] built upon Tripathy's GAN model[126] and

TABLE IX
SUMMARY OF ADVERSARIAL MACHINE LEARNING

Ref.	Publication	Scenarios	Target protected information	Model	Approach	Dataset
[116]	2018, USENIX security	CL	Attribute privacy	Multi-class logistic regression, Neural network	AttriGuard	Google+ dataset
[117]	2018, CCS	CL	Membership privacy	Alexnet, DenseNet, 4-layer FNN, 5-layer FNN	Min-max game	CIFAR100,Texas100, Purchase100
[118]	2019, CCS	CL	Membership privacy	6-layer FNN, 6-layer FNN, CNN	MemGuard	Location, Texas100, CH-MNIST
[119]	2020, TIFS	CL	Attribute privacy	DNN, DNN(resize), RAN	CPGAN	MNIST, HAR, GENKI-4K
[120]	2022, NeurIPS	FL	Data privacy	CNN	Contrastive adversarial learning	ADULT, NEWS, CelebA
[121]	2023, TIFS	CL	Attribute privacy	Logistic regression, DNN	Class-Overlapping	UTKFace, FairFace
[122]	2023, TIFS	CL	Model privacy	CNN	Adversarial perturbation	CIFAR10, GTSRB

introduced the concept of privacy-preserving class overlap in domain adaptation. They developed a privacy-preserving class overlap approach that employs a distribution-matching technique to train the privatizers. The Wasserstein distance[127] was utilized to measure the similarity between data distributions, thereby preventing attackers from accurately inferring sensitive attributes. Through experiments, they demonstrated that the proposed approach effectively preserves privacy, even when dealing with highly imbalanced datasets.

Similarly, Zhang et al. [122] conducted a study and observed that the confidence vector or the top-1 confidence obtained from the target model under attack (MUA) exhibited significant variations when queried with different inputs. Recognizing the potential risk of leaking rich internal information about the MUA and enabling model extraction attacks, they proposed a defense mechanism called adversarial confidence perturbation. To mitigate the risk, they employed an automatic optimization process wherein the MUA introduced subtle noise to each input query. This perturbation was designed to bring the model's confidence values closer to the decision boundary, thereby making it more challenging for attackers to gain insights into the model's internal information. Through experiments, they demonstrated the effectiveness of this approach, showing that it can reduce the accuracy of stolen models by up to 15%. By incorporating adversarial confidence perturbation, the proposed defense mechanism aims to enhance the security of MLaaS against model stealing attacks.

Recently, Qi et al. [120] proposed Contrastive Adversarial Learning in VFL to address privacy concerns. The main concept behind contrastive adversarial learning is to learn a protected representation that only contains information about a specific fairness-sensitive feature. This protected representation can then be shared with the platform, preserving the corresponding sensitive features while learning adversarial gradients without compromising user privacy. Experimental results on three datasets show that the proposed algorithm successfully protects user privacy with minimal loss of user information.

Comparisons and Summaries. Adversarial machine learning takes advantage of the attacker's strategies to enhance defense mechanisms. In this approach, the defender utilizes adversarial perturbations to obfuscate information or generate carefully crafted synthetic data to disrupt the attacker's efforts, ultimately aiming to strengthen the defense.

D. Other defense

With the continuous development of deep learning technology, there are now several new defense technologies emerging. With the continuous development of deep learning technology, there are now several new defense technologies emerging. In Table X, we list some of the more recent articles and classify them into two categories: Watermarking techniques and Vulnerability detection. In addition, we list publications, models, Defense against attacks, and datasets for these articles.

1) Watermarking techniques: As the size of deep learning models continues to grow, the computational resources and training data required for training these models have become increasingly expensive. Consequently, there is a potential for a lucrative business model involving the sale of pre-trained models. However, a significant challenge in this domain is the ease with which these models can be copied and redistributed. To address this issue, researchers have proposed watermarking techniques. These techniques aim to incorporate tracking mechanisms within the models, allowing them to be uniquely identified as the intellectual property of a specific vendor. By embedding watermarks, the origin and ownership of the models can be established, thus deterring unauthorized distribution and protecting the interests of the vendors.

The concept of watermarking, originally used for verifying ownership of digital media such as images, videos, or audio, has been adapted for deep neural networks. Nagai et al. [134] introduced this concept into the realm of deep learning by embedding watermarks into the parameters of the target model. They achieved this by incorporating a parameter regularizer during the training process. Building upon this work, Adi et al. [128] extended the watermarking technique to the black box setting, where the internal structure of the model is unknown. They leveraged the over-parameterization characteristic of neural networks to design a robust watermarking algorithm that could withstand various nuisances and retain

TABLE X
SUMMARY OF OTHER DEFENSE

Ref.	Publication	Scenarios	Model	Approach	Defense against attacks	Dataset
[128]	2018, USENIX Security	CL	DNN	Using overparameterization	Model extraction	CIFAR-10, CIFAR-100, ImageNet
[129]	2021, MM	CL	NN	Dynamic Adversarial Watermarking	Model extraction	MNIST, GTSRB, CIFAR10, Caltech256, ImageNet
[130]	2021, USENIX Security	CL	NN	Entangled Watermarking Embeddings	Model extraction	MNIST, Fashion-MNIST, CIFAR-10, Speech Commands
[131]	2022, S&P	CL	DNN	DEEPJUDGE	Model extraction	MNIST, CIFAR-10, ImageNet, SpeechCommands
[132]	2022, USENIX Security	CL	NN	Assessing Model Risk	Membership inference, Model inversion, Property inference, Model extraction	CelebA, Fashion-MNIST, STL10, UTKFace
[133]	2022, CCS	CL	NN	Automatically discover vulnerabilities	Property inference	Adult, Census, Insurance

its effectiveness even in scenarios where limited information about the target model is available.

In a similar vein, Jia et al. [130] proposed a technique called Entangled Watermarking Embeddings (EWE). EWE achieves entanglement between the data and the watermark by utilizing a feature that classifies the data sampled from the task distribution and the data encoded with the watermark. This entanglement ensures that deleting the watermark would require sacrificing performance. The authors trained the watermark task and the main task simultaneously, allowing the model to learn both tasks jointly. They demonstrated the effectiveness of EWE on four visual datasets and an audio dataset, showcasing its usability in various domains.

Nevertheless, one major concern with watermarking techniques is their intrusiveness, as they often require modifying the training process to embed the watermark. This can potentially compromise the utility of the model or introduce new security vulnerabilities. To address this, Chen et al. [108] proposed a multi-angle DNN copyright protection test framework. This framework provides a comprehensive characterization of the DNN model from various perspectives using multi-level test indicators. By employing this approach, non-invasive copyright detection is achieved, mitigating the limitations of single-feature detection and ensuring adaptability to the inherent randomness of DNN models.

2) Vulnerability detection: In addition to the aforementioned defense methods, an active defense method has recently been proposed. Liu et al. [132] introduced a holistic risk assessment method for machine learning models. This approach involves evaluating models prior to their release, allowing for the identification of potential security and privacy risks and preemptive measures to prevent attacks. Similarly, Cretu et al. [133] developed QuerySnout, a system designed to automatically discover vulnerabilities in query-based systems. QuerySnout operates as a black box, taking the target record as input. By analyzing the system's behavior on one or more datasets, it generates a multi-set query and combination rules to extract sensitive properties of the target record, thereby revealing potential vulnerabilities.

V. CHALLENGES AND FUTURE WORK

Based on the analysis of existing research, it has been observed that certain attacks rely on a white-box environment, which is often not present in real-world scenarios. Additionally, some defense methods impose significant computational and communication overhead, or may even impact the accuracy of the model. This paper aims to review inferential attack and defense techniques, systematically summarize the theoretical concepts and research progress about relevant attack and defense methods, and provide valuable references for future researchers to conduct further investigations. Furthermore, the upcoming section will explore and examine future research trends and challenges related to inferential attacks and their countermeasures.

Improve the robustness of attacks: Many attacks on machine learning models exploit the vulnerability of overfitting. Some attacks even require access to the internal details of the model, which can be challenging to achieve in real-world scenarios. As a response to overfitting attacks, various defense mechanisms have been developed, including differential privacy and adversarial attacks. So, it is essential to continue developing more powerful attacks that can overcome these defenses and explore new vulnerabilities in machine learning models.

Expand research on attack methods to include different models and domains: Research on attack methods should extend to encompass various models and domains in machine learning. While numerous models and domains, including classification models and generative models, have been targets of inference attacks, some emerging models and domains like self-supervised learning, meta-learning, and homogeneous federated learning have received less attention. Given the growing significance of these models in machine learning, investigating and developing inference attack strategies against them is crucial. This research will not only enhance privacy defenses but also contribute to the dynamic evolution of machine learning.

Tradeoff between privacy, efficiency, and accuracy: Existing privacy defense schemes face challenges in achieving a balance between privacy, efficiency, and accuracy. Tech-

niques such as homomorphic encryption or secure multi-party computation can provide strong privacy protection, but the computational overhead associated with complex encryption functions can significantly impact system performance. On the other hand, differential privacy offers improved efficiency, but the introduction of random noise can lead to a reduction in the accuracy of model predictions. Therefore, future work in privacy reasoning should focus on striking a balance between these three aspects. It is essential to develop novel techniques and methodologies that can effectively protect privacy while minimizing the impact on system efficiency and model accuracy.

Construct a perfect offensive and defensive system: The current defense mechanisms may not be sufficient to address future unknown attack threats. Relying solely on existing defense methods without proactive planning to counter new or specialized attack threats may result in inadequate security measures for technical products. Moreover, when a system faces security threats involving multiple attack types, traditional single solutions may not yield effective defense outcomes. In federated learning, which involves multiple participants, it is crucial to design personalized privacy protection mechanisms that cater to the specific privacy needs of different participants. This ensures that only necessary participants or specific attributes are protected, enhancing the efficiency and effectiveness of privacy measures. In the future, it is necessary to analyze and deduce all potential attack vectors and privacy issues based on existing attack methods and defense techniques. This analysis can be combined with secure encryption technologies to construct a robust security attack and defense system.

VI. Conclusion

Privacy inference attacks and defenses are related to user privacy. Therefore, this review focuses on the privacy problem of inferential attacks in machine learning, focusing on the basic concepts, action stages and related articles of various attacks from centralized learning and federated learning. At the same time, according to the different nature of the attack means, combined with the existing research work, several defense means are analyzed and summarized. Attack and defense is a synergistic relationship, a powerful means of attack will give birth to a targeted defense technology, and solid defense technology will promote the means of attack upgrade. More importantly, it introduces the challenges and future work of reasoning in attack and defense to advance research efforts in this area.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62206238, 62176122), the Natural Science Foundation of Jiangsu Province (No. BK20220562), the Natural Science Research Project of Universities in Jiangsu Province (No. 22KJB520010), the China Postdoctoral Science Foundation (No. 2023M732985).

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 770–778.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 4171–4186.

[3] F. Wu, X.-Y. Jing, P. Wei, C. Lan, Y. Ji, G.-P. Jiang, and Q. Huang, "Semi-supervised multi-view graph convolutional networks with application to webpage classification," *Inf. Sci.*, vol. 591, pp. 142–154, 2022.

[4] X. Chen, S. Xiang, C.-L. Liu, and C. Pan, "Vehicle detection in satellite images by hybrid deep convolutional neural networks," *IEEE Geosci. Remote. Sens. Lett.*, vol. 11, no. 10, pp. 1797–1801, 2014.

[5] G. L. Wittel and S. F. Wu, "On attacking statistical spam filters," in *CEAS 2004 - First Conference on Email and Anti-Spam, July 30-31, 2004, Mountain View, California, USA*, 2004.

[6] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 12:1–12:19, 2019.

[7] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, ser. Proceedings of Machine Learning Research, A. Singh and X. J. Zhu, Eds., vol. 54. PMLR, 2017, pp. 1273–1282.

[8] M. Ali, F. Naeem, M. Tariq, and G. Kaddoum, "Federated learning for privacy preservation in smart healthcare systems: A comprehensive survey," *IEEE J. Biomed. Health Informatics*, vol. 27, no. 2, pp. 778–789, 2023.

[9] W. Yang, Y. Zhang, K. Ye, L. Li, and C.-Z. Xu, "Ffd: A federated learning based method for credit card fraud detection," in *Big Data - BigData 2019 - 8th International Congress, Held as Part of the Services Conference Federation, SCF 2019, San Diego, CA, USA, June 25-30, 2019, Proceedings*, ser. Lecture Notes in Computer Science, K. Chen, S. Seshadri, and L.-J. Zhang, Eds., vol. 11514. Springer, 2019, pp. 18–32.

[10] A. Hard, K. Rao, R. Mathews, S. Ramaswamy, F. Beaufays, S. Augenstein, H. Eichner, C. Kiddon, and D. Ramage, "Federated learning for mobile keyboard prediction," *arXiv preprint arXiv:1811.03604*, 2018.

[11] L. Lyu, H. Yu, X. Ma, C. Chen, L. Sun, J. Zhao, Q. Yang, and P. S. Yu, "Privacy and robustness in federated learning: Attacks and defenses," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2022.

[12] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, pp. 619–640, 2021.

[13] A. Bhowmick, J. C. Duchi, J. Freudiger, G. Kapoor, and R. M. Rogers, "Protection against reconstruction and its applications in private federated learning," *ArXiv*, vol. abs/1812.00984, 2018.

[14] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2019, pp. 691–706.

[15] N. Agarwal, A. T. Suresh, F. X. Yu, S. Kumar, and B. McMa-

han, "cpsgd: Communication-efficient and differentially-private distributed sgd," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 7575–7586.

[16] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE Symposium on Security and Privacy (SP)*. San Jose, CA, USA: IEEE, May 2017, pp. 3–18.

[17] A. Pyrgelis, C. Troncoso, and E. D. Cristofaro, "Knock knock, who's there? membership inference on aggregate location data," in *Proceedings 2018 Network and Distributed System Security Symposium*. San Diego, CA: Internet Society, 2018.

[18] A. Sablayrolles, M. Douze, Y. Ollivier, C. Schmid, and H. Jégou, "White-box vs black-box: Bayes optimal strategies for membership inference," Aug. 2019.

[19] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "Ml-leaks: Model and data independent membership inference attacks and defenses on machine learning models," in *Proceedings 2019 Network and Distributed System Security Symposium*. San Diego, CA: Internet Society, 2019.

[20] L. Song, R. Shokri, and P. Mittal, "Privacy risks of securing machine learning models against adversarial examples," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. London United Kingdom: ACM, Nov. 2019, pp. 241–257.

[21] J. Hayes, L. Melis, G. Danezis, and E. De Cristofaro, "Logan: Membership inference attacks against generative models," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 1, pp. 133–152, Jan. 2019.

[22] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," in *2019 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2019, pp. 739–753.

[23] Y. Mao, X. Zhu, W. Zheng, D. Yuan, and J. Ma, "A novel user membership leakage attack in collaborative deep learning," in *2019 11th International Conference on Wireless Communications and Signal Processing (WCSP)*. Xi'an, China: IEEE, Oct. 2019, pp. 1–6.

[24] C. Song and A. Raghunathan, "Information leakage in embedding models," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event USA: ACM, Oct. 2020, pp. 377–390.

[25] Y. He, S. Rahimian, B. Schiele, and M. Fritz, "Segmentations-leak: Membership inference attacks and defenses in semantic image segmentation," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham: Springer International Publishing, 2020, vol. 12368, pp. 519–535.

[26] D. Chen, N. Yu, Y. Zhang, and M. Fritz, "Gan-leaks: A taxonomy of membership inference attacks against generative models," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event USA: ACM, Oct. 2020, pp. 343–362.

[27] K. Leino and M. Fredrikson, "Stolen memories: Leveraging model memorization for calibrated white-box membership inference," in *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 1605–1622.

[28] H. Liu, J. Jia, W. Qu, and N. Z. Gong, "Encodermi: Membership inference against pre-trained encoders in contrastive learning," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event Republic of Korea: ACM, Nov. 2021, pp. 2081–2095.

[29] M. Zhang, Z. Ren, Z. Wang, P. Ren, Z. Chen, P. Hu, and Y. Zhang, "Membership inference attacks against recommender systems," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event Republic of Korea: ACM, Nov. 2021, pp. 864–879.

[30] Z. Li and Y. Zhang, "Membership leakage in label-only exposures," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event Republic of Korea: ACM, Nov. 2021, pp. 880–895.

[31] S. Truex, L. Liu, M. E. Gursoy, L. Yu, and W. Wei, "Demystifying membership inference attacks in machine learning as a service," *IEEE Transactions on Services Computing*, vol. 14, no. 6, pp. 2073–2089, Nov. 2021.

[32] L. Song and P. Mittal, "Systematic evaluation of privacy risks of machine learning models," in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, Aug. 2021, pp. 2615–2632.

[33] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 3093–3106.

[34] Y. Liu, Z. Zhao, M. Backes, and Y. Zhang, "Membership inference attacks by exploiting loss trajectory," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 2085–2098.

[35] L. Watson, C. Guo, G. Cormode, and A. Sablayrolles, "On the importance of difficulty calibration in membership inference attacks," in *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[36] Y. Gu, Y. Bai, and S. Xu, "Cs-mia: Membership inference attack based on prediction confidence series in federated learning," *Journal of Information Security and Applications*, vol. 67, p. 103201, Jun. 2022.

[37] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *2022 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2022, pp. 1897–1914.

[38] X. Yuan and L. Zhang, "Membership inference attacks and defenses in neural network pruning," *USENIX Security Symposium*, p. 19, Feb. 2022.

[39] Z. Li, Y. Liu, X. He, N. Yu, M. Backes, and Y. Zhang, "Auditing membership leakages of multi-exit networks," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 1917–1931.

[40] G. Zhang, B. Liu, T. Zhu, M. Ding, and W. Zhou, "Label-only membership inference attacks and defenses in semantic segmentation models," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 2, pp. 1435–1449, Mar. 2023.

[41] H. Yan, S. Li, Y. Wang, Y. Zhang, K. Sharif, H. Hu, and Y. Li, "Membership inference attacks against deep learning models via logits distribution," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 5, pp. 3799–3808, Sep. 2023.

[42] Y. Liu, P. Jiang, and L. Zhu, "Subject-level membership inference attack via data augmentation and model discrepancy," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 5848–5859, 2023.

[43] N. Carlini, C. Liu, Ú. Erlingsson, J. Kos, and D. X. Song, "The secret sharer: Evaluating and testing unintended memorization in neural networks," in *USENIX Security Symposium*, 2018.

[44] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[45] M. Backes, P. Berrang, M. Humbert, and P. Manoharan, "Membership privacy in microrna-based studies," *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[46] X. He, R. Wen, Y. Wu, M. Backes, Y. Shen, and Y. Zhang, "Node-level membership inference attacks against graph neural networks," Feb. 2021.

[47] C. A. Choquette-Choo, F. Tramer, N. Carlini, and N. Papernot, "Label-only membership inference attacks," Dec. 2021.

[48] A. Shafran, S. Peleg, and Y. Hoshen, "Membership inference attacks are easier on difficult problems," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada: IEEE, Oct. 2021, pp. 14 800–14 809.

[49] L. Niu, M. S. Mirza, Z. Maradni, and C. Pöpper, "Codexleaks: Privacy leaks from code generation language models in github copilot," in *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, J. A. Calandrino and C. Troncoso, Eds. USENIX Association, 2023, pp. 2133–2150.

[50] Y. Shi and Y. E. Sagduyu, "Membership inference attack and defense for wireless signal classifiers with deep learning," *IEEE Transactions on Mobile Computing*, vol. 22, no. 7, pp. 4032–4043, Jul. 2023.

[51] N. Carlini, C. Liu, Erlingsson, J. Kos, and D. Song, "The secret sharer: Evaluating and testing unintended memorization in neural networks," in *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, N. Heninger and P. Traynor, Eds. USENIX Association, pp. 267–284.

[52] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. Dallas Texas USA: ACM, Oct. 2017, pp. 587–601.

[53] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. Oxford: IEEE, Jul. 2018, pp. 268–282.

[54] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, "Property inference attacks on fully connected neural networks using permutation invariant representations," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. Toronto Canada: ACM, Oct. 2018, pp. 619–633.

[55] Z. Yang, E.-C. Chang, and Z. Liang, "Adversarial neural network inversion via auxiliary knowledge alignment," no. arXiv:1902.08552, Feb. 2019.

[56] A. Salem, A. Bhattacharya, M. Backes, M. Fritz, and Y. Zhang, "Updates-leak: Data set inference and reconstruction attacks in online learning," in *29th USENIX Security Symposium, USENIX Security 2020, August 12-14, 2020*, S. Capkun and F. Roesner, Eds. USENIX Association, 2020, pp. 1291–1308.

[57] B. Jayaraman and D. Evans, "Are attribute inference attacks just imputation?" in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 1569–1582.

[58] X. Wang and W. H. Wang, "Group property inference attacks against graph neural networks," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 2871–2884.

[59] J. Zhou, Y. Chen, C. Shen, and Y. Zhang, "Property inference attacks against gans," in *Proceedings 2022 Network and Distributed System Security Symposium*. San Diego, CA, USA: Internet Society, 2022.

[60] S. Mahloujifar, E. Ghosh, and M. Chase, "Property inference from poisoning," in *2022 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2022, pp. 1120–1137.

[61] S. Mehnaz, S. V. Dibbo, E. Kabir, N. Li, and E. Bertino, "Are your sensitive attributes private? novel model inversion attribute inference attacks on classification models," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 4579–4596.

[62] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: Information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. Dallas Texas USA: ACM, Oct. 2017, pp. 603–618.

[63] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. Paris, France: IEEE, Apr. 2019, pp. 2512–2520.

[64] Z. He, T. Zhang, and R. B. Lee, "Model inversion attacks against collaborative inference," in *Proceedings of the 35th Annual Computer Security Applications Conference*. San Juan Puerto Rico USA: ACM, Dec. 2019, pp. 148–162.

[65] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 14 747–14 756.

[66] B. Zhao, K. R. Mopuri, and H. Bilen, "idlg: Improved deep leakage from gradients," Jan. 2020.

[67] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients - how easy is it to break privacy in federated learning?" in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, Virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin, Eds., 2020.

[68] X. Luo, Y. Wu, X. Xiao, and B. C. Ooi, "Feature inference attack on model predictions in vertical federated learning," in *2021 IEEE 37th International Conference on Data Engineering (ICDE)*. Chania, Greece: IEEE, Apr. 2021, pp. 181–192.

[69] C. Fu, X. Zhang, S. Ji, J. Chen, J. Wu, S. Guo, J. Zhou, A. X. Liu, and T. Wang, "Label inference attacks against vertical federated learning," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 1397–1414.

[70] B. Wang and N. Z. Gong, "Stealing hyperparameters in machine learning," in *2018 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA: IEEE, May 2018, pp. 36–52.

[71] T. Orekondy, B. Schiele, and M. Fritz, "Knockoff nets: Stealing functionality of black-box models," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA: IEEE, Jun. 2019, pp. 4949–4958.

[72] H. Yu, K. Yang, T. Zhang, Y.-Y. Tsai, T.-Y. Ho, and Y. Jin, "Cloudleak: Large-scale deep learning models stealing through adversarial examples," in *Proceedings 2020 Network and Distributed System Security Symposium*. San Diego, CA: Internet Society, 2020.

[73] S. Pal, Y. Gupta, A. Shukla, A. Kanade, S. Shevade, and V. Ganapathy, "Activethief: Model extraction using active learning and unannotated public data," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 865–872, Apr. 2020.

[74] V. Chandrasekaran, K. Chaudhuri, I. Giacomelli, S. Jha, and S. Yan, "Exploring connections between active learning and model extraction," in *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 1309–1326.

[75] B. G. Atli, S. Szyller, M. Juuti, S. Marchal, and N. Asokan, "Extraction of complex dnn models: Real threat or boogeyman?" in *Engineering Dependable and Secure Machine Learning Systems*, O. Shehory, E. Farchi, and G. Barash, Eds. Cham: Springer International Publishing, 2020, vol. 1272, pp. 42–57.

[76] M. Jagielski, N. Carlini, D. Berthelot, A. Kurakin, and N. Pa-

pernot, "High accuracy and high fidelity extraction of neural networks," in *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Aug. 2020, pp. 1345–1362.

[77] Z. Yue, Z. He, H. Zeng, and J. McAuley, "Black-box attacks on sequential recommenders via data-free model extraction," in *Fifteenth ACM Conference on Recommender Systems*. Amsterdam Netherlands: ACM, Sep. 2021, pp. 44–54.

[78] S. Z. Béguelin, S. Tople, A. Paverd, and B. Köpf, "Grey-box extraction of natural language models," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, M. Meila and T. Zhang, Eds., vol. 139. PMLR, 2021, pp. 12 278–12 286.

[79] A. Dziedzic, N. Dhawan, M. A. Kaleem, J. Guan, and N. Papernot, "On the difficulty of defending self-supervised learning against model extraction," in *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvári, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 2022, pp. 5757–5776.

[80] A. S. Rakin, M. H. I. Chowdhuryy, F. Yao, and D. Fan, "Deepsteal: Advanced model extractions leveraging efficient weight stealing in memories," in *2022 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2022, pp. 1157–1174.

[81] Q. Pan, J. Wu, A. K. Bashir, J. Li, and J. Wu, "Side-channel fuzzy analysis-based ai model extraction attack with information-theoretic perspective in intelligent iot," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 11, pp. 4642–4656, Nov. 2022.

[82] J. B. Gray, "Introduction to linear regression analysis," *Technometrics*, vol. 44, pp. 191 – 192, 2002.

[83] J. R. Correia-Silva, R. Berriel, C. S. Badue, A. F. de Souza, and T. Oliveira-Santos, "Copycat cnn: Stealing knowledge by persuading confession with random non-labeled data," *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, 2018.

[84] H. Hu, Z. Salcic, L. Sun, G. Dobbie, and X. Zhang, "Source inference attacks in federated learning," in *2021 IEEE International Conference on Data Mining (ICDM)*. Auckland, New Zealand: IEEE, Dec. 2021, pp. 1102–1107.

[85] A. Gadotti and F. Houssiau, "Pool inference attacks on local differential privacy: Quantifying the privacy guarantees of apple's count mean sketch in practice," *USENIX Security Symposium*, p. 19, 2022.

[86] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. Vienna Austria: ACM, Oct. 2016, pp. 308–318.

[87] N. Phan, M. N. Vu, Y. Liu, R. Jin, D. Dou, X. Wu, and M. T. Thai, "Heterogeneous gaussian mechanism: Preserving differential privacy in deep learning with provable robustness," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, Jul. 2019, pp. 4753–4759.

[88] D. Chen, T. Orekondy, and M. Fritz, "Gs-wgan: A gradient-sanitized approach for learning differentially private generators," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, Virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M.-F. Balcan, and H.-T. Lin, Eds., 2020.

[89] T. Cao, A. Bie, A. Vahdat, S. Fidler, and K. Kreis, "Don't generate me: Training differentially private generative models with sinkhorn divergence," in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, Virtual*, M. Ranzato, A. Beygelzimer, Y. N. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021, pp. 12 480–12 492.

[90] H. Yan, X. Li, H. Li, J. Li, W. Sun, and F. Li, "Monitoring-based differential privacy mechanism against query flooding-based model extraction attack," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 4, pp. 2680–2694, Jul. 2022.

[91] D. Ye, S. Shen, T. Zhu, B. Liu, and W. Zhou, "One parameter defense—defending against data inference attacks via differential privacy," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1466–1480, 2022.

[92] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. Denver Colorado USA: ACM, Oct. 2015, pp. 1310–1321.

[93] M. A. Heikkilä, E. Lagerspetz, S. Kaski, K. Shimizu, S. Tarkoma, and A. Honkela, "Differentially private bayesian learning on distributed data," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 3226–3235.

[94] R. C. Geyer, T. Klein, and M. Nabi, "Differentially private federated learning: A client level perspective," Mar. 2018.

[95] B. Jayaraman, L. Wang, D. Evans, and Q. Gu, "Distributed learning without distress: Privacy-preserving empirical risk minimization," in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., 2018, pp. 6346–6357.

[96] L. Zhao, Q. Wang, Q. Zou, Y. Zhang, and Y. Chen, "Privacy-preserving collaborative deep learning with unreliable participants," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1486–1500, 2020.

[97] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. S. Quek, and H. V. Poor, "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.

[98] Q. Wang, H. Yin, T. Chen, J. Yu, A. Zhou, and X. Zhang, "Fast-adapting and privacy-preserving federated recommender system," *The VLDB Journal*, vol. 31, no. 5, pp. 877–896, Sep. 2022.

[99] M. Nasr, S. Songi, A. Thakurta, N. Papernot, and N. Carlin, "Adversary instantiation: Lower bounds for differentially private machine learning," in *2021 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2021, pp. 866–882.

[100] J. Liu, M. Juuti, Y. Lu, and N. Asokan, "Oblivious neural network predictions via minionn transformations," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. Dallas Texas USA: ACM, Oct. 2017, pp. 619–631.

[101] P. Mohassel and Y. Zhang, "Secureml: A system for scalable privacy-preserving machine learning," in *2017 IEEE Symposium on Security and Privacy (SP)*. San Jose, CA, USA: IEEE, May 2017, pp. 19–38.

[102] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth, "Practical secure aggregation for privacy-preserving machine learning," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. Dallas Texas USA: ACM, Oct. 2017, pp. 1175–1191.

[103] P. Mohassel and P. Rindal, "Aby ^{3}: A mixed protocol frame-

work for machine learning," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. Toronto Canada: ACM, Oct. 2018, pp. 35–52.

[104] N. Agrawal, A. Shahin Shamsabadi, M. J. Kusner, and A. Gascón, "Quotient: Two-party secure neural network training and prediction," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. London United Kingdom: ACM, Nov. 2019, pp. 1231–1247.

[105] D. Rathee, M. Rathee, N. Kumar, N. Chandran, D. Gupta, A. Rastogi, and R. Sharma, "Cryptflow2: Practical 2-party secure inference," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. Virtual Event USA: ACM, Oct. 2020, pp. 325–342.

[106] C. Guo, J. Katz, X. Wang, and Y. Yu, "Efficient and secure multiparty computation from fixed-key block ciphers," in *2020 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA, USA: IEEE, May 2020, pp. 825–841.

[107] Z. Huang, W.-j. Lu, C. Hong, and J. Ding, "Cheetah: Lean and fast secure two-party deep neural network inference," in *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, Aug. 2022, pp. 809–826.

[108] H. Chen, H. Li, Y. Wang, M. Hao, G. Xu, and T. Zhang, "Privdt: An efficient two-party cryptographic framework for vertical decision trees," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 1006–1021, 2023.

[109] M. Hao, H. Li, H. Chen, P. Xing, and T. Zhang, "Fastsecnet: An efficient cryptographic framework for private neural network inference," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 2569–2582, 2023.

[110] X. Jiang, M. Kim, K. Lauter, and Y. Song, "Secure outsourced matrix computation and application to neural networks," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. Toronto Canada: ACM, Oct. 2018, pp. 1209–1222.

[111] H. Chen, W. Dai, M. Kim, and Y. Song, "Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. London United Kingdom: ACM, Nov. 2019, pp. 395–412.

[112] C. Zhang, S. Li, J. Xia, W. Wang, F. Yan, and Y. Liu, "Batchcrypt: Efficient homomorphic encryption for cross-silo federated learning," in *2020 USENIX Annual Technical Conference, USENIX ATC 2020, July 15-17, 2020*, A. Gavrilovska and E. Zadok, Eds. USENIX Association, 2020, pp. 493–506.

[113] N. Samardzic, A. Feldmann, A. Krastev, S. Devadas, R. Dreslinski, C. Peikert, and D. Sanchez, "F1: A fast and programmable accelerator for fully homomorphic encryption," in *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*. Virtual Event Greece: ACM, Oct. 2021, pp. 238–252.

[114] K. Cong, D. Das, J. Park, and H. V. Pereira, "Sortinghat: Efficient private decision tree evaluation via homomorphic encryption and transciphering," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 563–577.

[115] D. Kim and C. Guyot, "Optimized privacy-preserving cnn inference with fully homomorphic encryption," *IEEE Trans. Inf. Forensics Secur.*, vol. 18, pp. 2175–2187, 2023.

[116] J. Jia and N. Z. Gong, "Attriguard: A practical defense against attribute inference attacks via adversarial machine learning," in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018*, W. Enck and A. P. Felt, Eds. USENIX Association, 2018, pp. 513–529.

[117] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. Toronto Canada: ACM, Oct. 2018, pp. 634–646.

[118] J. Jia, A. Salem, M. Backes, Y. Zhang, and N. Z. Gong, "Memguard: Defending against black-box membership inference attacks via adversarial examples," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. London United Kingdom: ACM, Nov. 2019, pp. 259–274.

[119] B.-W. Tseng and P.-Y. Wu, "Compressive privacy generative adversarial network," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2499–2513, 2020.

[120] T. Qi, F. Wu, C. Wu, L. Lyu, T. Xu, H. Liao, Z. Yang, Y. Huang, and X. Xie, "Fairvfl: A fair vertical federated learning framework with contrastive adversarial learning," in *NeurIPS*, 2022.

[121] T.-H. Lin, Y.-S. Lee, F.-C. Chang, J. M. Chang, and P.-Y. Wu, "Protecting sensitive attributes by adversarial training through class-overlapping techniques," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1283–1294, 2023.

[122] J. Zhang, S. Peng, Y. Gao, Z. Zhang, and Q. Hong, "Apmsa: Adversarial perturbation against model stealing attacks," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1667–1679, 2023.

[123] T. Orekondy, B. Schiele, and M. Fritz, "Towards a visual privacy advisor: Understanding and predicting privacy risks in images," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 3706–3715.

[124] M. S. Ryoo, B. Rothrock, C. Fleming, and H. J. Yang, "Privacy-preserving human activity recognition from extreme low resolution," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, S. Singh and S. Markovitch, Eds. AAAI Press, 2017, pp. 4255–4262.

[125] C. Song and V. Shmatikov, "Overlearning reveals sensitive attributes," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[126] A. Tripathy, Y. Wang, and P. Ishwar, "Privacy-preserving adversarial networks," in *57th Annual Allerton Conference on Communication, Control, and Computing, Allerton 2019, Monticello, IL, USA, September 24-27, 2019*. IEEE, 2019, pp. 495–505.

[127] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 214–223.

[128] Y. Adi, C. Baum, M. Cissé, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *27th USENIX Security Symposium, USENIX Security 2018, Baltimore, MD, USA, August 15-17, 2018*, W. Enck and A. P. Felt, Eds. USENIX Association, 2018, pp. 1615–1631.

[129] S. Szyller, B. G. Atli, S. Marchal, and N. Asokan, "Dawn: Dynamic adversarial watermarking of neural networks," in *Proceedings of the 29th ACM International Conference on Multimedia*. Virtual Event China: ACM, Oct. 2021, pp. 4417–4425.

[130] H. Jia, C. A. Choquette-Choo, V. Chandrasekaran, and N. Papernot, "Entangled watermarks as a defense against model extraction," in *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, M. Bailey and R. Greenstadt, Eds. USENIX Association, 2021, pp. 1937–1954.

[131] J. Chen, J. Wang, T. Peng, Y. Sun, P. Cheng, S. Ji, X. Ma, B. Li, and D. Song, "Copy, right? a testing framework for copyright protection of deep learning models," in *2022 IEEE Symposium on Security and Privacy (SP)*. San Francisco, CA,

USA: IEEE, May 2022, pp. 824–841.

[132] Y. Liu, R. Wen, X. He, A. Salem, Z. Zhang, M. Backes, E. D. Cristofaro, M. Fritz, and Y. Zhang, "Ml-doctor: Holistic risk assessment of inference attacks against machine learning models," in *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, K. R. B. Butler and K. Thomas, Eds. USENIX Association, 2022, pp. 4525–4542.

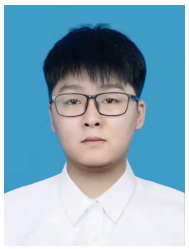
[133] A.-M. Cretu, F. Houssiau, A. Cully, and Y.-A. de Montjoye, "Querysnout: Automating the discovery of attribute inference attacks against query-based systems," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*. Los Angeles CA USA: ACM, Nov. 2022, pp. 623–637.

[134] Y. Nagai, Y. Uchida, S. Sakazawa, and S. Satoh, "Digital watermarking for deep neural networks," *Int. J. Multim. Inf. Retr.*, vol. 7, no. 1, pp. 3–16, 2018.

Bosen Rao received the B.S. degree from Yangzhou University, Yangzhou, China, in 2023. He is currently working toward the M.S. degree in computer science and technology with Yangzhou University, Yangzhou, China. His research interests include federated learning and privacy inference attacks.

Jiale Zhang (Member, IEEE) received the Ph.D. degree in computer science and technology the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2021. He is currently an Associate Professor with the School of Information Engineering, Yangzhou University, Yangzhou, China. He has published over 40 research papers in refereed international conferences and journals, such as IEEE TII, IEEE IoT-J, COSE, JSS, IEEE ICC, and IEEE Globecom. His research interests are mainly federated learning, AI security, and blockchian security.

Di Wu (Member, IEEE) is a Lecturer at the School of Mathematics, Physics, and Computing, the University of Southern Queensland and a Visiting Fellow at School of Computer Science, University of Technology Sydney. Prior to that, he was a Researcher at the Australian Institute for Machine Learning (AIML) School of Computer Science, University of Adelaide, Adelaide, Australia. Previous to this, he was an Associate Research Fellow, Artificial Intelligence at Deakin Blockchain Innovation Lab, School of Information Technology, Deakin University, Melbourne, Australia, and worked as a Postdoc Fellow at the School of Computer Science, University of Technology Sydney (UTS), Sydney, Australia. He has more than 10 years of experience in research development and academia. He has substantial industry experience in large project management, software development, and large system maintenance experience while working on various projects at China Telecom (Global 500), Shanghai. His research area focuses on applying AI on edge devices and AI applications. He has published over 20 papers in refereed books, conferences, and journals. He has served as a special session chair for IJCNN. He also serves as a reviewer for many high-quality academic conferences and journals, such as CoRL, PR, TETCI, and so on.



Chengcheng Zhu received the B.S. degree from Yangzhou University, Yangzhou, China, in 2022. He is currently working toward the M.S. degree in computer science and technology with Yangzhou University, Yangzhou, China. His research interests include backdoor attacks and adversarial learning.

Xiaobing Sun (Member, IEEE) received the B.S. degree in computer science and technology from Jiangsu University of Science and Technology, Zhenjiang, China, in 2007, and the Ph.D. degree from the School of Computer Science and Engineering, Southeast University, Nanjing, China, in 2012. He is currently a Professor with the School of Information Engineering, Yangzhou University, Yangzhou, China. His research interests include software maintenance and evolution, software repository mining, and intelligence analysis. He has been authorized more than 20 patents. He has published more than 80 papers in refereed international journals.

Bing Chen (Member, IEEE) is currently a full professor at the College of Compute Science and Technology, Nanjing University of Aeronautics and Astronautics (NUAA), China. He received his B.S. and M.S. in Computer Engineering from NUAA in 1992 and 1995, and the Ph.D. degrees from the College of Compute Science and Technology, NUAA, in 2008. His research interests include cloud/edge computing, security and privacy, FL, wireless communications.