FedIPR: Ownership Verification for Federated Deep Neural Network Models

Bowen Li, Lixin Fan, *Member, IEEE*, Hanlin Gu,
Jie Li, *Senior Member, IEEE*, and Qiang Yang, *Fellow, IEEE*

Abstract—Federated learning models are collaboratively developed upon valuable training data owned by multiple parties. During the development and deployment of federated models, they are exposed to risks including illegal copying, re-distribution, misuse and/or free-riding. To address these risks, the ownership verification of federated learning models is a prerequisite that protects federated learning model intellectual property rights (IPR) i.e., FedIPR. We propose a novel federated deep neural network (FedDNN) ownership verification scheme that allows private watermarks to be embedded and verified to claim legitimate IPR of FedDNN models. In the proposed scheme, each client independently verifies the existence of the model watermarks and claims respective ownership of the federated model without disclosing neither private training data nor private watermark information. The effectiveness of embedded watermarks is theoretically justified by the rigorous analysis of conditions under which watermarks can be privately embedded and detected by multiple clients. Moreover, extensive experimental results on computer vision and natural language processing tasks demonstrate that varying bit-length watermarks can be embedded and reliably detected without compromising original model performances. Our watermarking scheme is also resilient to various federated training settings and robust against removal attacks.

Index Terms—Model IPR protection, ownership verification, federated learning, model watermarking, backdoor training

1 Introduction

The successful applications of deep neural network (DNN) to computer vision, natural language processing and data mining tasks come at the cost of the expensive training process: a) the training incurs substantial efforts and costs in terms of expertise, dedicated hardware, and exceedingly long time for the designing and training DNN *models*; b) it requires a vast amount of training *data* to boost the model performance, which often increases monotonically with the volume of training data [1], [2], [3], [4]. To protect both the valuable training data and the trained DNN models from being illegally copied, re-distributed or misused, therefore, becomes a compelling need that motivates our research work reported in this article.

To protect the Intellectual Property Rights (IPR) of Deep Neural Network, DNN watermarking techniques have been proposed in [5], [6], [7], [8], [9], [10], [11], [12] to embed designated watermarks into DNN models. Subsequently, DNN ownership is verified by robustly extracting the embedded watermarks from the model in question. Note that both *feature-based watermarks*[9], [10], [11] and *backdoor-based watermarks*[12] have been proposed to verify ownership of DNN models. In order to protect valuable training data in a collaborative learning setting whereas *semi-honest* adversaries may attempt to espy participants' private information, a secure federated learning (SFL) framework has been proposed[13], [14], [15] to collaboratively train a federated deep neural network (FedDNN) without giving away to adversaries private training data[16] and data feature distribution [17]. Therefore, each client in federated learning must a) *not* disclose to other parties any information about private training data; and b) prove ownership of the trained model without disclosing their private watermarks. The first requirement has been fulfilled by protecting the exchanged local models using techniques such as homomorphic encryption (HE) [18], differential privacy (DP)[19] or secret sharing[20] albeit at cost of degraded model performances [21]. The second requirement is one of the open problems considered in this work.

Taking into consideration threat models in both DNN watermarking and secure federated learning, we propose a unified framework called FedIPR which consists of two separate processes along with standard SFL learning procedures: a) *a watermark embedding process* that allows multiple parties to embed their secret feature-based and backdoor-based watermarks; b) *a verification process* that allows each party to independently verify the ownership of FedDNN model.

- *Bowen Li and Jie Li are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China. E-mail: {li-bowen, lijiecs}@sjtu.edu.cn.*
- *Lixin Fan and Hanlin Gu are with the WeBank AI Lab, WeBank, Shenzhen 518000, China. E-mail: {Lixin.Fan01, ghltsl123}@gmail.com.*
- *Qiang Yang is with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Hong Kong, and also with WeBank AI Lab, WeBank, Shenzhen 518000, China. E-mail: qyang@cse.ust.hk.*

Manuscript received 23 March 2022; revised 15 July 2022; accepted 30 July 2022. Date of publication 2 August 2022; date of current version 6 March 2023.
This work was supported in part by the National Key Research and Development Program of China under Grants 2020YFB1805501, 2018AAA0101100 and 2020YFB1806700, in part by NSFC under Grant 61932014, and in part by the Key Research and Development Program of Jiangsu, China under Grant BE2020026.
(Corresponding author: Lixin Fan.)
Recommended for acceptance by M. Mahoney.
This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2022.3195956, provided by the authors.
Digital Object Identifier no. 10.1109/TPAMI.2022.3195956

0162-8828 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.
Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

Two technical challenges for embedding watermarks into FedDNN model are investigated in this paper:

挑战1:
各个client的
watermark不
会互相干扰

- *Challenge A: how to ensure that private watermarks embedded by different clients into the same FedDNN model do not discredit each other?* This challenge is unique in a federated learning setting whereas different client's watermarks may potentially conflict with each other (see Fig. 3 for an example). As a solution to the challenge, theoretical analysis in Theorem 1 elucidates conditions under which multiple feature-based watermarks can be embedded into the same FedDNN model without bringing each other into discredit, and based on the theoretical analysis, a feature-based watermarking method dedicated for horizontal federated learning is proposed. (see Section 5.3 for details).
- *Challenge B: how to ensure that embedded watermarks are robust to privacy-preserving learning strategies?* This challenge is due to modifications of model parameters brought by various privacy preserving methods e.g., differential privacy [19], defensive aggregation[22], [23], [24] and client selection[13]. As a solution, FedIPR adopts robust client-side training to embed both feature-based and backdoor-based watermarks. Our empirical results in Section 6 show that robust feature-based and backdoor-based watermarks are persistent under various federated learning strategies.

挑战2:
保证watermark
不会因为隐私
增强技术而失
效

Moreover, extensive experiments on computer vision and natural language processing tasks demonstrate that feature-based watermarks embedded in *normalization scale parameters* (see Section 5.3 for details) are highly reliable, while backdoor-based watermarks can be reliably detected for black-box ownership verification. In short, main contributions of our work are threefold:

- We put forth the first general framework called FedIPR for ownership verification of DNN models in a secure federated learning setting. FedIPR is designed in such a way that each client can embed his/her own private feature-based and backdoor-based watermarks and verify watermarks to claim ownership independently.
- We demonstrate successful applications of FedIPR for various DNN model architectures trained in the semi-honest federated learning setting. Theoretical analysis of the *significance* of feature-based watermarks and superior performance with extensive experimental results showcase the efficacy of the proposed FedIPR framework.
- FedIPR also provides an effective method to detect freeriders[25], [26] who do not contribute data or computing resources but participate in federated learning to get for free the valuable model. Due to the lack of rightful watermarks embedded in the FedDNN model, freeriders can be discerned from benign participants.

To our best knowledge, the FedIPR framework is the first technical solution that supports the protection of DNN ownerships in a secure federated learning setting such that secret watermarks embedded in FedDNN models do not disclose to semi-honest adversaries.

The rest of the paper is organized as follows: Section 2 briefly reviews previous work related to secure federated learning and DNN ownership verification. Section 3 describes the preliminary background for FedIPR. Section 4 illustrates the proposed FedIPR framework formulation. Section 5 delineates the watermark embedding approaches both in white-box and black-box modes, and Section 6 presents experimental results and showcases the robustness of FedIPR. We discuss and conclude the paper in Section 7.

2 Related Work

We briefly review related work in three following aspects and refer readers to survey articles in respective aspects[15], [27], [28].

2.1 Secure Federated Learning

Secure Federated learning[13], [14], [15] aims to collaboratively train a global machine learning model among multiple clients without disclosing private training data to each other [18], [19], [20], [29]. Moreover, privacy-preserving techniques such as homomorphic encryption [18], differential privacy [19] and secret sharing [20] were often used to protect exchanged local models[13], [14].

2.2 Threats to Model IPR

It was shown that FedDNN models of high commercial values were subjected to severe IPR threats[25], [30], [31], [32]. First, unauthorized parties might plagiarize the DNN model with non-technical methods[30]. Second, Tramer et al. showcased *model stealing attacks* that aimed to steal deployed victim models even if attackers have no knowledge of training samples or model parameters[31]. Third, Fraboni et al. demonstrated that *freeriders* might join in federated learning and plagiarized the valuable models with no real contributions to the improvements of federated models[25].

2.3 DNN Watermarking Methods

As a counter measure against model plagiarisms, private watermarks are embedded into the DNN model parameters and functionality, which have been strongly combined with the protected DNN model. Two categories of DNN watermarking methods have been proposed:

Backdoor-based methods proposed to use a particular set of inputs as the triggers and let the model deliberately output specific incorrect labels[12], [33], [34]. Backdoor-based methods collected evidence of suspected plagiarism through remote API without accessing internal parameters of models. We also refer to a recent survey[28] for more existing watermark embedding schemes A survey on model watermarking neural networks

Feature-based methods proposed to encode designated binary strings as watermarks into layer parameters in DNN models[5], [9], [10], [11], [35]. Specifically, Uchida et al. [9] proposed to embed feature-based watermarks into convolution layer weights using a binary cross-entropy loss function. Fan et al. [36] proposed to embed feature-based watermarks into *normalization layer scale parameters* of the convolution block with a hinge-like regularization term. In the verification stage of feature-based watermarks, one must access DNN internal parameters to detect watermarks.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

For federated learning model verification scenario, double masking protocols[37], [38] are proposed as FedDNN integrity verification schemes while guaranteeing user's privacy in the training process. However, those model integrity verification methods could not preserve the IPR of FedDNN models. For IPR protection of FedDNN, Atli et al. [39] adopted *backdoor-based* watermarks to enable ownership verification for the central server. Nevertheless, they only considered the setting in which the server was responsible for embedding watermarks into the global FedDNN model, and did not allow clients to embed and verify private watermarks. Liu et al. [40] has adopted client-side backdoor-based watermarking method under the homomorphic encryption FL framework, while our proposed FedIPR consider both feature-based and backdoor-based watermarking in a general secure federated learning scenario with strategies like differential privacy[19], homomorphic encryption[18], defensive aggregation[22], etc.

3 Preliminaries

In this section, we first review and formulate key ingredients of the secure horizontal federated learning and existing DNN watermarking methods as follows. We also explain in Table 1 all notations used in this article.

3.1 Secure Horizontal Federated Learning

A secure horizontal federated learning[14] system consists of K clients which build local models[1] with their own data and send local models $\{\mathbf{W}_k\}_{k=1}^K$ to an aggregator to obtain a global model. The aggregator conducts the following aggregation process[13], [14], [15]

$$\mathbf{W} \leftarrow \sum_{k=1}^{K} \frac{n_k}{K} \mathbf{W}_k, \tag{1}$$

where n_k is the weight for each client's local model $\mathbf{W}_k$.

Remark. in secure federated learning, local model $\mathbf{W}_k$ might be protected by using Homomorphic Encryption (HE)[18], Differential Privacy (DP)[19] such that semi-honest adversaries can not infer private information from $\mathbf{W}_k$. These privacy preserving strategies pose one challenge to be addressed for reliable watermarking (see Section 4.3).

3.2 Freeriders in Federated Learning

In federated learning, there might be *freerider* clients[25] who do not contribute data or computing resources but construct some superficial local models to participate in training only to obtain the global model for free. Specifically, there are several strategies for freeriders to construct local models[25]:

Freeriding With Previous Models (Plain Freerider). Freeriders create a superficial model as follows[25] 仅仅把上轮的global model贡献给server

$$\mathbf{W}^{free} = Free(\mathbf{W}^t, \mathbf{W}^{t-1}), \tag{2}$$

in which $\mathbf{W}^t$, $\mathbf{W}^{t-1}$ denote respectively local models from two previous iterations. Note that the construction of this superficial model costs nothing for freeriders since they are merely saved copies of model parameters from previous iterations.

TABLE 1
Notations Used in This Article

Notations	Descriptions
K	Number of clients in Secure Federated Learning
$\mathbb{N}$	Federated neural network model
$\mathbf{W}$	Model weights of model $\mathbb{N}$
$\mathbf{W}_k$	Local model of kth client $\mathbb{N}$
$\mathcal{G}()$	Key **Generation** Process
$N_\mathbf{T}$	Bit-length of backdoor-based watermarks
$\mathbf{T}$	Target *backdoor-based* watermarks
$(\mathbf{X_T}, \mathbf{Y_T})$	Samples and labels of backdoor-based watermarks $\mathbf{T}$
N	Bit-length of feature-based watermarks
$\mathbf{B}$	Target *feature-based* watermarks
$\tilde{\mathbf{B}}$	*Feature-based* watermarks extracted from the parameters
$\theta = \{\mathbf{S}, \mathbf{E}\}$	Secret parameters for *feature-based* watermarks
$\mathbf{S}$	Watermark location parameters
$\mathbf{E}$	Watermark embedding matrix
$\mathcal{E}()$	Watermark **Embedding** Process
$\mathbf{W}_k^t$	Local model of kth client at communication round t
$\mathbf{W}^t$	Global model at communication round t
L_D	The loss function for the main learning task
$L_\mathbf{T}$	Backdoor-based watermark embedding regularization term
$L_{\mathbf{B},\theta}$	Feature-based watermark embedding regularization term
$\mathcal{A}()$	**Aggregation** Process in Secure Federated Learning
$\mathcal{V}()$	Watermark **Verification** Process
$\mathcal{V}_W()$	White-box verification
$\mathcal{V}_B()$	Black-box verification
η_F	Detection rate of feature-based watermarks
η_T	Detection rate of backdoor-based watermarks

Freeriding With Gaussian Noise. Freeriders adopt the previous global model parameters $\mathbf{W}^{t-1}$ and add Gaussian noise to simulate a local model 加noise 后贡献给server

$$\mathbf{W}^{free} = \mathbf{W}^t + \xi_t, \quad \xi_t \sim \mathcal{N}(0, \sigma_t). \tag{3}$$

Detection methods are proposed to detect and eliminate superficial local models as such[26]. However it is required to train a meta freerider detector. 插值呢? 顺序打乱呢?

3.3 DNN Watermarking Methods

There are broadly two categories of DNN watermarking methods: 特点：可黑箱，客戶端验证，基于验证数据在训练阶段嵌入

Backdoor-Based Watermarks[12], [34]. Backdoor-based watermarks $\mathbf{T} = \{(\mathbf{X}_\mathbf{T}^1, \mathbf{Y}_\mathbf{T}^1), \ldots, (\mathbf{X}_\mathbf{T}^{N_\mathbf{T}}, \mathbf{Y}_\mathbf{T}^{N_\mathbf{T}})\}$ are embedded into the model function $\mathbb{N}$ during the training time by incorporating a loss function of backdoor samples.

In the verification step (as shown in the second procedure of Fig. 1), backdoor samples are used as the trigger input to the model $\mathbb{N}$. The ownership is successfully verified if the detection error of designated backdoor labels is less than a threshold ϵ_B 用特定样本上的输出结果作为验证方式

$$\mathcal{V}_B(\mathbb{N}, \mathbf{T}) = \begin{cases} \text{TRUE}, & \text{if } \mathbb{E}_{\mathbf{T}_n}(\mathbb{I}(\mathbf{Y_T} \neq \mathbb{N}(\mathbf{X_T}))) \leq \epsilon_B, \\ \text{FALSE}, & \text{otherwise}, \end{cases} \tag{4}$$

in which $\mathcal{V}_B()$ is the ownership verification process that only accesses model API in black-box mode.

Remark. $(\mathbf{B}, \boldsymbol{\theta}, \mathbf{T})$ are private watermarking information that should be kept secret without disclosing to other parties.

[1] Other works[13] also call them model updates, because the local models are equal to model updates for aggregation.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

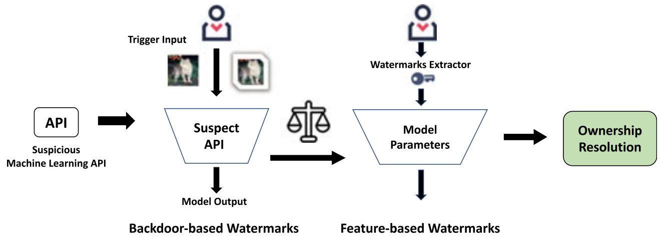


Fig. 1. Ownership verification processes composed of backdoor-based watermarks and feature-based watermarks

特点：需白箱，在训练阶段以约束条件嵌入，需要extractor，需要具备norm layer

Feature-Based Watermarks [9], [10], [11], [36]. In the watermark embedding step, N-bits target binary watermarks $\mathbf{B} \in \{0,1\}^N$ are embedded during the learning of model parameters $\mathbf{W}$, by adding regularization terms to the original learning task.

During the verification step (as shown in the third procedure of Fig. 1), feature-based watermarks $\tilde{\mathbf{B}}$ extracted with extractor $\boldsymbol{\theta}$ from DNN parameters is then matched with the designated watermarks $\mathbf{B}$, to judge if Hamming distance $\mathrm{H}(\mathbf{B}, \tilde{\mathbf{B}})$ is less than a preset threshold ϵ_W

$$\mathcal{V}_W(\mathbf{W}, (\mathbf{B}, \boldsymbol{\theta})) = \begin{cases} \text{TRUE}, & \text{if } \mathrm{H}(\mathbf{B}, \tilde{\mathbf{B}}) \leq \epsilon_W, \\ \text{FALSE}, & \text{otherwise}, \end{cases} \tag{5}$$

in which $\mathcal{V}_W()$ is the ownership verification process that has to access model parameters in a white-box mode.

4 Federated DNN Ownership Verification

We propose a novel watermark embedding and ownership verification scheme called FedIPR for the secure horizontal federated learning scenario. FedIPR is designed in the way such that each client can a) protect his/her private data; and b) embed and verify his/her own private watermarks without disclosing information about private watermarks.

4.1 FedIPR: FedDNN Ownership Verification with Watermarks

Following the framework of SFL in Section 3.1, we give below a formal definition of the FedIPR ownership verification scheme, which is pictorially illustrated in Fig. 2.

Definition 1. *A Federated Deep Neural Network (FedDNN) model ownership verification scheme (FedIPR) for a given network $\mathbb{N}[]$ is defined as a tuple $\mathcal{V} = (\mathcal{G}, \mathcal{E}, \mathcal{A}, \mathcal{V}_W, \mathcal{V}_B)$ of processes, consisting of:*

I) *For client $k \in \{1, \ldots K\}$, a client-side* key generation process $\mathcal{G}() \rightarrow (\mathbf{B}_k, \theta_k, \mathbf{T}_k)$ *generates target watermarks $\mathbf{B}_k$, watermark extraction parameters $\theta_k = \{\mathbf{S}_k, \mathbf{E}_k\}$* and *a* trigger set *(backdoor-based watermarks) $\mathbf{T}_k = \{(\mathbf{X}^1_{\mathbf{T}_k}, \mathbf{Y}^1_{\mathbf{T}_k}), \ldots, (\mathbf{X}^{N_\mathbf{T}}_{\mathbf{T}_k}, \mathbf{Y}^{N_\mathbf{T}}_{\mathbf{T}_k})\}$;*

   Remark. *the $(\mathbf{B_k}, \theta_\mathbf{k}, \mathbf{T_k})$ are private watermarking parameters that should be kept secret without disclosing to other clients. In the extraction parameters $\theta_k = \{\mathbf{S}_k, \mathbf{E}_k\}$, $\mathbf{S}_k$ denotes the location of watermarks $\mathbf{B}_k$, and $\mathbf{E}_k$ denotes the secret embedding matrix for watermarks $\mathbf{B}_k$.*

II) *A client-side FedDNN* embedding process $\mathcal{E}()$ *minimizes the combined loss L_k of the main task , and two regularization terms $L_{\mathbf{T}_k}$ and $L_{\mathbf{B}_k, \theta_k}$ to embed trigger samples $\mathbf{T}_k$ and feature-based watermarks $\mathbf{B}_k$ respectively*[2]*, once receives the global model $\mathbf{W}^t$ at communication round t*

$$L_k := \underbrace{L_{D_k}(\mathbf{W}^t)}_{\text{main task}} + \alpha_k \underbrace{L_{\mathbf{T}_k}(\mathbf{W}^t)}_{\text{backdoor-based}} + \beta_k \underbrace{L_{\mathbf{B}_k, \theta_k}(\mathbf{W}^t)}_{\text{feature-based}}, \quad k \in \{1, \ldots K\}, \tag{6}$$

where D_k denotes the training data of client k, α_k[3] *denotes the parameters to control the backdoor-based watermarking loss $L_{\mathbf{T}_k}$, and β_k denotes the factor for feature-based watermarking regularization term $L_{\mathbf{B}_k, \theta_k}$.*

   Remark. *Note that a* $\mathrm{ClientUpdate}(L_k, \mathbf{W}^t) =:$ *$argminL_k$ sub-routine seeks the optimal parameters and sends local model to the aggregator (see below for a server-side aggregation process).*

III) *A server-side FedDNN* aggregation process $\mathcal{A}()$ *collects local models from m randomly selected clients and performs model aggregation using the* `FedAvg` *algorithm [13] i.e.,*

$$\mathbf{W}^{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} \mathbf{W}_k^{t+1}, \tag{7}$$

where $\mathbf{W}_k^{t+1} \leftarrow \mathrm{ClientUpdate}(L_k, \mathbf{W}^t)$ is the local model of client k at round t, and $\frac{n_k}{n}$ denoted the aggregation weight for `Fedavg` *algorithm.*

   Remark: *in SFL, strategies like Differential Privacy [42], defensive aggregation mechanism[22], [23], [24] and client selection[13] are widely used for privacy, security and efficiency.*

   After the global model $\mathbf{W}$ is trained with convergence, each client can conduct ownership verification as follows.

IV) *A client-side* black-box verification process $\mathcal{V}_B()$ *checks whether the detection error of designated labels $\mathbf{Y}_{\mathbf{T}_k}$ generated by trigger samples $\mathbf{X}_{\mathbf{T}_k}$ is smaller than ϵ_B*

$$\mathcal{V}_B(\mathbb{N}, \mathbf{T}_k) = \begin{cases} \text{TRUE}, & \text{if } \mathbb{E}_{\mathbf{T}_k}(\mathbb{I}(\mathbf{Y}_{\mathbf{T}_k} \neq \mathbb{N}(\mathbf{X}_{\mathbf{T}_k}))) \leq \epsilon_B, \\ \text{FALSE}, & \text{otherwise}, \end{cases} \tag{8}$$

in which $\mathbb{I}()$ is the indicator function and $\mathbb{E}$ is the expectation over trigger set $\mathbf{T}_k$.

V) *A client-side* white-box verification process $\mathcal{V}_W()$ *extracts feature-based watermarks $\tilde{\mathbf{B}}_k = sgn(\mathbf{W}, \theta_k)$ with sign function $sgn()$ from the global model parameters $\mathbf{W}$, and verifies the ownership as follows:*

$$\mathcal{V}_W(\mathbf{W}, (\mathbf{B}_k, \theta_k)) = \begin{cases} \text{TRUE}, & \text{if } \mathrm{H}(\mathbf{B}_k, \tilde{\mathbf{B}}_k) \leq \epsilon_W, \\ \text{FALSE}, & \text{otherwise}, \end{cases} \tag{9}$$

in which $\mathrm{H}(\mathbf{B}_k, \tilde{\mathbf{B}}_k)$ is the Hamming distance between $\tilde{\mathbf{B}}_k$) and the target watermarks $\mathbf{B}_k$, and ϵ_W is a preset threshold.

随机确定alpha和beta

[2] A client k may opt-out and not embed watermarks or trigger samples by setting $\alpha_k = 0.0$ or $\beta_k = 0.0$. Following [41], we adopt a *random sampling* strategy in experiments to assign non-zero values to α_k, β_k to simulate the situation that clients make decisions on their own.

[3] If backdoor samples are filled in the implementation side of backdoor-based watermarking task, the parameter α_k is equal to 1

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

两种方式都需要client在训练阶段添加watermark，而且必须由client校验

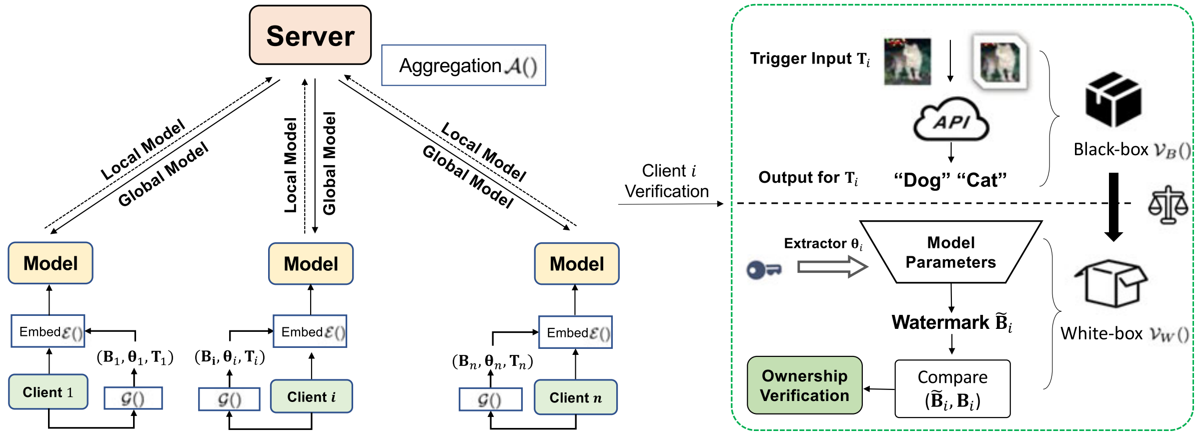

Server
Aggregation $\mathcal{A}()$
Local Model
Global Model
Local Model
Global Model
Local Model
Global Model
Model
Model
Model
Embed $\mathcal{E}()$
Embed $\mathcal{E}()$
Embed $\mathcal{E}()$
$(\mathbf{B}_1, \boldsymbol{\theta}_1, \mathbf{T}_1)$
$(\mathbf{B}_i, \boldsymbol{\theta}_i, \mathbf{T}_i)$
$(\mathbf{B}_n, \boldsymbol{\theta}_n, \mathbf{T}_n)$
Client 1
$\mathcal{G}()$
$\mathcal{G}()$
Client i
$\mathcal{G}()$
Client n
Client i Verification
Trigger Input $\mathbf{T}_i$
API
Output for $\mathbf{T}_i$
"Dog" "Cat"
Black-box $\mathcal{V}_B()$
Extractor θ_i
Model Parameters
Watermark $\tilde{\mathbf{B}}_i$
White-box $\mathcal{V}_W()$
Ownership Verification
Compare $(\tilde{\mathbf{B}}_i, \mathbf{B}_i)$

Fig. 2. An illustration of federated DNN (FedDNN) watermark Embedding and Verification scheme. Private watermarks are generated and embedded into the local models, which are then aggregated using the `FedAvg` algo. (the left panel). In case the federated model is plagiarized, each client may invoke verification processes to extract watermarks from the plagiarized model in both black-box and white-box manner to claim his/her ownership of the federated model (the right panel).

Watermark Detection Rate. For client ownership verification, the *watermark detection rate* can be defined as:

- For N bit-length feature-based watermarks $\mathbf{B}$, detection rate η_F is calculated as

模型一致的概率

$$\eta_F := 1 - \frac{1}{N} H(\mathbf{B}, \tilde{\mathbf{B}}), \tag{10}$$

where $H(\mathbf{B}, \tilde{\mathbf{B}})$ measures Hamming distance between extracted binary watermark string $\tilde{\mathbf{B}}$ and the target watermarks $\mathbf{B}$;

- For backdoor-based watermarks $\mathbf{T}$, the detection rate is

$$\eta_T := \mathbb{E}_{\mathbf{T}}(\mathbb{I}(\mathbf{Y_T} = \mathbb{N}(\mathbf{X_T}))), \tag{11}$$

which is calculated as the ratio of backdoor samples that are classified as designated labels w.r.t. the total number $N_{\mathbf{T}}$ of trigger set.

Given the definition of FedIPR, let us proceed to illustrate two technical challenges to be addressed by FedIPR.

添加feature-based watermark所加的额外约束条件是冲突的

4.2 Challenge A: Conflicting Goals of More than One Watermarks in FedDNN

For all clients, the *watermark capacity* measures the overall bit-length of watermarks that can be significantly verified. The first challenge is to determine the maximal capability of multiple watermarks that can be embedded by K clients in SFL without discrediting each other.

Specifically for *feature-based* watermarks, it remains an open question whether there is a common solution for different clients to embed their private designated watermarks. To illustrate the potential conflict between multiple watermarks, let us investigate the following examples:

Example 1. two clients need to embed different watermarks $\mathbf{B}_1 = 010$ and $\mathbf{B}_2 = 101$, respectively, into the same parameters $\mathbf{W} = (w_1, w_2, w_3, w_4, w_5)$ with the same embedding matrix

$$\mathbf{E} = \begin{pmatrix} e_{11} & e_{12} & e_{13} & e_{14} & e_{15} \\ e_{21} & e_{22} & e_{23} & e_{24} & e_{25} \\ e_{11} & e_{12} & e_{33} & e_{34} & e_{35} \end{pmatrix}^T, \tag{12}$$

the watermarks extracted from parameters $\mathbf{W} = (w_1, w_2, w_3, w_4, w_5)$ are $\tilde{\mathbf{B}} = \mathbf{WE}$, and regularization terms are used to constrain the parameters to satisfy

$$\begin{cases} (010): & \sum_{i=1}^{5} w_i e_{1i} < 0, \sum_{i=1}^{5} w_i e_{2i} > 0, \sum_{i=1}^{5} w_i e_{3i} < 0, \\ (101): & \sum_{i=1}^{5} w_i e_{1i} > 0, \sum_{i=1}^{5} w_i e_{2i} < 0, \sum_{i=1}^{5} w_i e_{3i} > 0, \end{cases} \tag{13}$$

it is obvious that two different watermarks again impose conflicting constraints that cannot be simultaneously satisfied by the same global model parameters.

General Case. for the feature-based watermarks $\{(\mathbf{B}_k, \boldsymbol{\theta}_k)\}_{k=1}^{K}$ embedded into the same model parameters $\mathbf{W}$ by K different clients, each client k embeds N bit-length of feature-based watermarks $\mathbf{B}_k = (t_{k1}, t_{k2}, \dots, t_{kN}) \in \{+1, -1\}^N$, the extracted watermarks $\tilde{\mathbf{B}}_k = \mathbf{WE}_k$ should be consistent with targeted watermarks $\mathbf{B}_k$, i.e.,

$$\forall j \in \{1, 2, \dots, N\} \quad and \quad k \in K, t_{kj}(\mathbf{WE}_k)_j > 0. \tag{14}$$

As Fig. 3 shows, each client k tries to guide the target parameter $\mathbf{W}_k$ to a special direction conditioned on $\mathbf{B}_k$, but each $\mathbf{W}_k$ is aggregated into an unified $\mathbf{W}$ according to Eq. (7) in SFL, these constraints may conflict with each other.

Theorem 1 elucidates the condition under which a feasible solution exists for K different watermarks to be embedded without conflicts and provides the lower bound of detection rate η_F.

Theorem 1. *For K different watermarks (N bit-length each) to embed in M channels of the global model parameters $\mathbf{W}$, take their detection rate to be measured as $\eta_F = 1 - \frac{1}{N} H(\mathbf{B}, \tilde{\mathbf{B}})$, where $H(\mathbf{B}, \tilde{\mathbf{B}})$ is the hamming distance between extracted watermarks $\tilde{\mathbf{B}}$ and the target watermarks $\mathbf{B}$. The watermark detection rate η_F satisfies:*

Case 1. *If $KN \leq M$ [4], then there exists $\mathbf{W}$ such that $\eta_F = 1$.*

[4] Another condition for this theorem is the embedding matrix $\mathbf{E}$ clients decide needs to be column (line) non-singular matrix (see the detail in Appendix A, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2022.3195956)

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

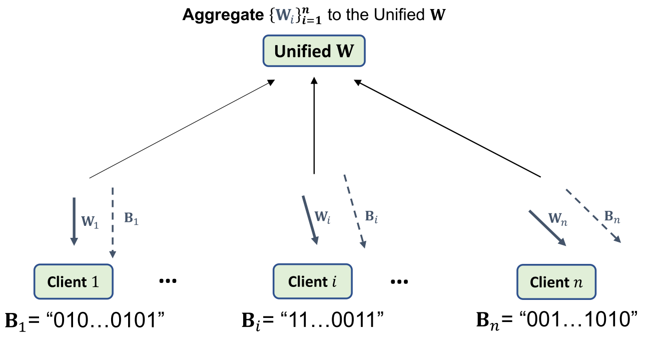

Aggregate $\{\mathbf{W}_i\}_{i=1}^{n}$ to the Unified W
Unified W
$\mathbf{W}_1$
$\mathbf{B}_1$
$\mathbf{W}_i$
$\mathbf{B}_i$
$\mathbf{W}_n$
$\mathbf{B}_n$
Client 1
…
Client i
…
Client n
$\mathbf{B}_1$= "010…0101"
$\mathbf{B}_i$= "11…0011"
$\mathbf{B}_n$= "001…1010"

Fig. 3. Different clients in federated learning adopt different regularization terms to embed feature-based watermarks

Case 2. *If $KN > M$, then, there exists* $\mathbf{W}$ *such that*

$$\eta_F \geq \frac{KN + M}{2KN}. \tag{15}$$

The proof is deferred in Appendix A, available in the online supplemental material.

Remark. results in *case 1* ($KN < M$) demonstrate the existence of a solution for watermark embedding when the total bit-length of all clients' watermarks KN is smaller than the total number M[5] of network channels that can be used to embed watermarks; results in *case 2* ($KN > M$) provides a lower bound of detection rate of embedded watermarks. For example, if $K = 10$ clients decide to each embed $N = 100$ bit-length watermarks into $M = 600$ channels of model parameters $\mathbf{W}$, the lower bound of detection rate is $\eta_F = 0.8$.

Taking both cases into account, we give below the optimal bit-length assigned to multiple watermarks under different situations (detailed analysis is deferred to Appendix B, available in the online supplemental material).

Optimal bit-length and maximal bit-length N[6]. We treat feature-based ownership verification as a hypothesis testing, where $\mathcal{H}_0$ is "the model is not plagiarized" versus $\mathcal{H}_1$ "model is plagiarized", and we get p-value as the statistical significance of watermarks, the upper bound is given

$$p\text{-}value \leq \sum_{i=\eta_F N}^{N} \binom{N}{i} (1/C)^i (1 - 1/C)^{N-i}, \tag{16}$$

in which $C = 2$.

With analysis of p-value (see detailed analysis in Appendix B, available in the online supplemental material), the optimal bit-length for the smallest p-value is $N_{opt} = M/K$, which is determined by the "smallest p-value" i.e., the most significant watermark verification.

In case that a p-value is only required to be less than a given statistical significance level α e.g., 0.0001, we can determine the range of bit-length N (N_{max} and N_{min}) that can guarentee that p-value is lower than the given level α.

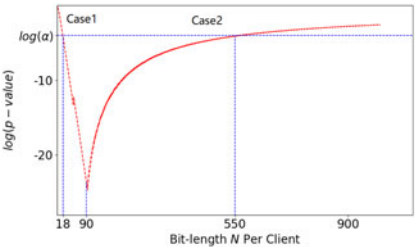


Fig. 4. *The optimal bit-length* and *the acceptable range of watermark bit-length* that provide strong confidence of ownership verification. As shown in figure, $K = 10, M = 896$, and $\eta_F = 0.98$ (we take by default) in case 1, the optimal bit-length for p-value is $N_{opt} = M/K = 90$, for an acceptable level $\alpha = 0.0001$, the acceptable range of watermark bit-length is from $N = 18$ to $N = 550$.

Take an example as in Fig. 4, if $K = 10$ clients decide to embed watermarks into $M = 896$ channels of model parameters, $N_{opt} = 90$, when $\alpha = 0.0001$, the corresponding N_{max} is 550 and N_{min} is 18. 但是这受限于客户端数量，不能适用于query-based

In short, analysis of Theorem 1 specifies *the optimal bit-length* and *the acceptable range of watermark bit-length* that allows reliable detection rates with sufficient significance to support ownership verification. Moreover, it is shown that empirical results in Section 6 are in accordance with the analysis elucidated in this section.

4.3 Challenge B: Robustness of Watermarks in FedDNN

The *robustness* of watermarks indicates whether the detection rate is persistent against various *training strategies* and *attacks* that attempt to remove the watermarks.

We investigate the robustness of both the feature-based and backdoor-based watermarks in the FedDNN model. Particularly, we measure the detection rate and statistical significance of watermarks with/without training strategies and attacks to report the robustness. The measurement settings including impacting factors (training strategies and attacks), targeted watermarks (feature-based watermarks in *Normalization parameters* $\mathbf{W}_\gamma$ and convolution parameters $\mathbf{W}_C$ and backdoor-based watermarks), and metrics are summarized in Table 2:

Training Strategies. In SFL, strategies like differential privacy[42], defensive aggregation mechanism[22], [23], [24] and client selection[13] are widely used for privacy, security and efficiency. Those training strategies modify the training processes of SFL, which may affect the detection rate i.e., the significance of watermarks:

- To protect data privacy, *differential privacy* mechanism[19] in SFL add noise to the local model of each client.
- For defending the model poisoning attack[22], [43], *defensive aggregation*[22], [23], [24] in SFL perform detect and filter some local models from each client.
- For communication efficiency, in each communication round, the server adopts certain *client selection* strategies[13] to pick up a random subset of clients and have their local models aggregated. Other clients do not need to upload local models until they are selected in future communication rounds.

Removal Attack. The attacker that steals the model may try to remove the watermarks while inheriting most model

[5] For example, $M = 896$ channels across the last 3 layers for AlexNet, $M = 2048$ channels across the last 4 layers for ResNet18 and $M = 2304$ channels across the last 3 layers for DistilBERT (illustrated in Appendix C, available in the online supplemental material).

[6] However, other factors may influence η_F. For example, as experiment results in Fig. 12, 10 demonstrated, random noise added to the federated learning process or removal attacks launched by plagiarizers, all conspire to degrade η_F to various extents.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
Reported Investigation Setting of Robustness

Impacting factor	Watermarks		Metrics	
	feature-based	backdoor-based	Detection Rate	p-value
Differential Privacy	$\mathbf{W}_\gamma$	✓	✓	✓
Client Selection	$\mathbf{W}_\gamma$	✓	✓	✓
Defensive Aggregation	$\mathbf{W}_\gamma$	✓	✓	✓
Pruning Attack	$\mathbf{W}_\gamma$, $\mathbf{W}_C$	✓	✓	
Fine-tuning Attack	$\mathbf{W}_\gamma$, $\mathbf{W}_C$	✓	✓	

performance. Following previous DNN watermarking methods[5], [7], [12], we investigate watermark robustness under *fine-tuning* and *pruning* attacks (see Algorithm 1 for pseudocodes).

Algorithm 1. Removal Attack

```
Input: Model $\mathbb{N}$, pruning rate $p$, additional training data $D_{add}$.
procedure PRUNING
  Pruning the model $\mathbb{N}$ with $p$ pruning rate.
procedure FINETUNING
  for epochs in 50 do
    Train the model $\mathbb{N}$ only in main classification task with additional training data $D_{add}$.
```

Pruning 后再训练50epochs

We adopt client-side watermark embedding method to investigate the watermark robustness. Extensive experimental results in Section 6.5 show that both backdoor-based and feature-based watermarks can be reliably detected with high significance (p-value less than $2.89e^{-15}$ is guaranteed).

5 Implementation of FedDNN Ownership Verification

Section 4 proposes the FedIPR framework, which allows clients to independently embed secret watermarks into the model and verify whether the designated watermarks exist in the model in question. We illustrate below a specific implementation of FedIPR framework with algorithm pseudocodes given in Algorithms 2, 3 and 4.

5.1 Watermark Generation

For a client k in the federated learning system, the watermarks adopted to mark trained FedDNN model include backdoor-based watermarks $\mathbf{T}_k$ and feature-based watermarks ($\mathbf{B}_k$, θ_k). Those watermarks are initialized and kept in secret as shown in Algorithm 2.

Algorithm 2. Generation $\mathcal{G}()$ of Watermarks

```
procedure WATERMARK GENERATION
  for client $k$ in $K$ clients do
    Initialize $(\mathbf{B}_k, \theta_k) = \{\mathbf{S}_k, \mathbf{E}_k\}$.
    Encode $\mathbf{B}_k$ into binary string.
    Initialize $\mathbf{T}_k = \{(\mathbf{X}^1_{\mathbf{T}_k}, \mathbf{Y}^1_{\mathbf{T}_k}), \ldots, (\mathbf{X}^{N_\mathbf{T}}_{\mathbf{T}_k}, \mathbf{Y}^{N_\mathbf{T}}_{\mathbf{T}_k})\}$.
  return$\{(\mathbf{B}_k, \theta_k, \mathbf{T}_k)\}^{k=K}_{k=1}$
```

As illustrated in Algorithm 3, the client-side FedDNN *embedding process* $\mathcal{E}()$ minimizes the weighted combined loss of main task and two regularization terms to embed watermarks $\mathbf{T}_k$ and $\mathbf{B}_k$ respectively.

Algorithm 3. Embedding Process $\mathcal{E}()$ of Watermarks

```
Each client k with its own watermark tuple $(\mathbf{B}_k, \theta_k, \mathbf{T}_k)$
for communication round $t$ do
  The server distributes the global model parameters $\mathbf{W}^t$ to each clients and randomly selects $cK$ out of $K$ clients.
  Local Training:
  for $k$ in selected $cK$ of $K$ clients do
    Sample mini-batch of $m$ training samples $\mathbf{X}\{\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(m)}\}$ and targets $\mathbf{Y}\{\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(m)}\}$.
    if Enable backdoor-based watermarks then
      Sample $t$ samples $\{\mathbf{X}^{(1)}_{\mathbf{T}_k}, \ldots, \mathbf{X}^{(t)}_{\mathbf{T}_k}\}$ , $\{\mathbf{Y}^{(1)}_{\mathbf{T}_k}, \ldots, \mathbf{Y}^{(t)}_{\mathbf{T}_k}\}$ from trigger set $(\mathbf{X}_{\mathbf{T}_k}, \mathbf{Y}_{\mathbf{T}_k})$
      Concatenate $\mathbf{X}$ with $\{\mathbf{X}^{(1)}_{\mathbf{T}_k}, \ldots, \mathbf{X}^{(t)}_{\mathbf{T}_k}\}$ , $\mathbf{Y}$ with $\{\mathbf{Y}^{(1)}_{\mathbf{T}_k}, \ldots, \mathbf{Y}^{(t)}_{\mathbf{T}_k}\}$.
    Compute cross-entropy loss $L_c$ using $\mathbf{X}$ and $\mathbf{Y}$
    ▷ Batch poisoning approach is adopted, thus $\alpha_l = 1$, $L_c = L_{D_K} + L_{\mathbf{T}_k}$.
    for layer $l$ in targeted layers set $\mathsf{L}$ do
      Compute feature-based regularization term $L^l_{\mathbf{B}_k,\theta_k}$ using $\theta_k$ and $\mathbf{W}^l$
    $L_{\mathbf{B}_k,\theta_k} \leftarrow \sum_{l \in \mathsf{L}} L^l_{\mathbf{B}_k,\theta_k}$
    $L_k = L_c + \beta_k L_{\mathbf{B}_k,\theta_k}$
    Backpropagate using $L_k$ and update $\mathbf{W}^t_k$
  Server Update:
  Aggregate local models $\{\mathbf{W}^t_k\}^K_{k=1}$ with FedAvg algorithm
```

先backdoor再featrue

5.2 Backdoor-Based Watermark Embedding and Verification

To embed backdoor-based watermarks $\mathbf{T}_k = \{(\mathbf{X}^1_{\mathbf{T}_k}, \mathbf{Y}^1_{\mathbf{T}_k}), \ldots, (\mathbf{X}^{N_\mathbf{T}}_{\mathbf{T}_k}, \mathbf{Y}^{N_\mathbf{T}}_{\mathbf{T}_k})\}$, the model owner trains the model with an additional backdoor training task where the loss function of backdoor training $L_\mathbf{T}(\mathbf{W}^t)$ is defined with cross entropy (CE) loss

$$L_\mathbf{T}(\mathbf{W}^t) = CE(\mathbf{Y}_{\mathbf{T}_k}, \mathbb{N}(\mathbf{X}_{\mathbf{T}_k})). \tag{17}$$

Adversarial Samples as Triggers. In our FedIPR scheme, we adopt adversarial samples as the triggers. Basically, adversarial samples $(\mathbf{X}_\mathbf{T}, \mathbf{Y}_\mathbf{T})$ are generated from original data $(\mathbf{X}, \mathbf{Y})$ with Projected Gradient Descent (PGD)[44]. Our backdoor-based watermark scheme adopts those backdoor samples as the trigger input during both training time and inference time.

Each backdoor-based watermarks $\mathbf{X}_\mathbf{T}$ is verified, provided that for the input $\mathbf{X}_\mathbf{T}$, the model outputs the designated label, i.e., $\mathbb{N}(\mathbf{X}_\mathbf{T}) = \mathbf{Y}_\mathbf{T}$.

5.3 Feature-Based Watermark Embedding and Verification

In FedIPR, each client k chooses its own $(\mathbf{B}_k, \boldsymbol{\theta}_k)$ as the feature-based watermarks, which are embedded with a regularization term $L_{\mathbf{B}_k,\theta_k}$ along with main task loss.

Approach of FedIPR. FedIPR proposes that each client embeds its own watermarks $\mathbf{B}_k$ with secret parameters $\theta_k = (\mathbf{S}_k, \mathbf{E}_k)$:

$$L_{\mathbf{B}_k,\theta_k}(\mathbf{W}^t) = L_{\mathbf{B}_k}(\mathbf{S}_k, \mathbf{W}^t, \mathbf{E}_k), \tag{18}$$

whereas the secret watermarking parameters $\theta_k = (\mathbf{S}_k, \mathbf{E}_k)$ are only known for client k, and FedIPR proposes to embed the watermarks into the *normalization layer scale parameters*

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

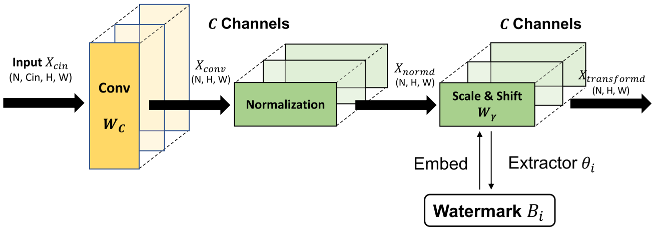


Fig. 5. Layer structure of a convolution layer: normalization layer weights $\mathbf{W}_\gamma$ (in green) are used to embed watermarks, and the watermarks extracted in a white-box manner with a secret embedding matrix.

of the convolution block, i.e., $\mathbf{S}_k(\mathbf{W}) = \mathbf{W}_\gamma = \{\gamma_1, \ldots, \gamma_C\}$, where C is the number of normalization channels in $\mathbf{W}_\gamma$.

We adopt a secret embedding matrix $\mathbf{E}_k \in \theta_k = (\mathbf{S}_k, \mathbf{E}_k)$ to embed and extract watermarks, the distance between targeted watermarks and extracted watermarks is implemented as following regularization term:

$$\begin{aligned} L_{\mathbf{B}_k,\theta_k}(\mathbf{W}^t) &= L_{\mathbf{B}_k}\left(\mathbf{W}^t_\gamma \mathbf{E}_k, \mathbf{B}_k\right) \\ &= \mathrm{HL}\left(\mathbf{B}_k, \tilde{\mathbf{B}}_k\right) = \sum_{j=1}^{N} \max(\mu - b_j t_j, 0), \end{aligned} \tag{19}$$

where we note $\tilde{\mathbf{B}}_k = \mathbf{W}^t_\gamma \mathbf{E}_k$ as extracted watermarks and we implement the regularization term as *hinge-like* loss HL() on the target watermarks $\mathbf{B}_k = (t_1, \ldots, t_N) \in \{0,1\}^N$ and extracted watermarks $\tilde{\mathbf{B}}_k = (b_1, \ldots, b_N) \in \{0,1\}^N$, and μ is the parameter of hinge loss.

In SFL, we implement feature-based watermark embedding for two different network architectures including convolution neural network (CNN) and transformer-based neural network.

如果没有norm layer呢？用norm stat的好处是不会受到DNN交换不变性质的影响

5.3.1 Feature-Based Watermarks in CNN

As illustrated in Fig. 5, for a convolution neural network $\mathbb{N}(\mathbf{W})$, we may choose the convolution kernel weights $\mathbf{W}_C$ or the normalization layer weights $\mathbf{W}_\gamma$ to embed feature-based watermarks.

Fan et al. [5], [36] has reported that normalization layer weights $\mathbf{W}_\gamma = (\gamma_1, \ldots, \gamma_C) \in \{-1,+1\}^C$ are suitable model parameters to embed robust binary watermark strings as follows:

$$O(x_i^p) = \gamma_i * x_i^p + \beta_i, \tag{20}$$

in which x_i^p is the model parameters in channel i and β_i is the offset parameter of normalization (see [5], [36] and Appendix C, available in the online supplemental material for details).

In FedIPR, we choose $\mathbf{W}_\gamma$ to embed feature-based watermarks for robust performance, which is reported in Section 6. For ablation study, we compare the robustness of watermarks in the normalization layers and convolution layers in Fig. 12, the results show that watermarks in the normalization layers are more persistent against removal attacks.

5.3.2 Feature-Based Watermarks in Transformer-Based Networks

As illustrated in Fig. 6, feature-based watermarks can also be applied to transformer-based network. A transformer encoder block[45] is organized with a *Layer-Normalization layer* (the mean output value is normalized in the channel direction, which has an obvious effect for accelerating the convergence performance). The same to normalization in CNN, the process is controlled by parameters $\mathbf{W}_\gamma$ in the *Layer-Normalization layer*.

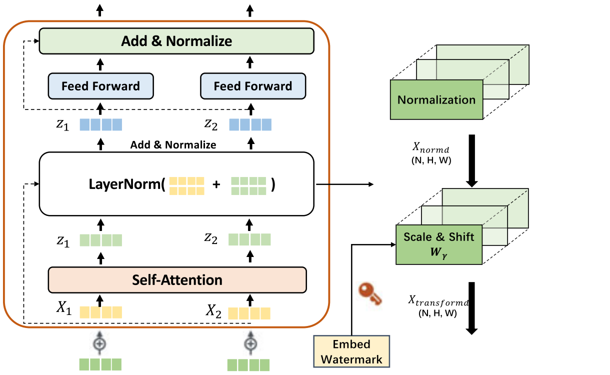


Fig. 6. Layer structure of an encoder block: normalization layer weights $\mathbf{W}_\gamma$ (in green) are used to embed feature-based watermarks which are extracted in white-box manner.

总之，依赖特定的网络结构用于嵌入

In FedIPR, we choose $\mathbf{W}_\gamma$ to embed the feature-based watermarks, such that the watermarks can be persistent in the model architecture.

5.4 Ownership Verification with Watermarks

Once the model is plagiarized by unauthorized party, the model owner can call experiments of ownership verification. As Algorithm 4 and Fig. 1 illustrates, for the backdoor-based watermark verification, the model owner query the API with designed triggers, and for feature-based verification, it detected the feature-based watermarks embedded in the normalization layer according to the sign of according channel parameters. Combined with both backdoor-based and feature-based watermarks, the statistical significance of ownership verification is obtained.

Algorithm 4. Ownership Verification $\mathcal{V}_B()$ and $\mathcal{V}_W()$

```
Input: API $\mathbb{N}()$ offered by adversaries, Triggers $(\mathbf{X_T}, \mathbf{Y_T})$ provided by owner; Model weights $\mathbf{W}$ of model, secret parameters $\boldsymbol{\theta} = (\mathbf{S}, \mathbf{E})$ and target watermarks $\mathbf{B}$ provided by user.
1: procedure WATERMARK DETECTION
2:     Input the backdoor $\mathbf{X_T}$ into model $\mathbb{N}$ to derive the classification label $\mathbb{N}(\mathbf{X_T})$
3:     Match $\mathbb{N}(\mathbf{X_T})$ with target backdoor label $\mathbf{Y_T}$
4:     Compute the backdoor detection rate $\eta_T = \mathcal{V}_B(\mathbf{X_T}, \mathbf{Y_T}, \mathbb{N})$
5:     $\tilde{\mathbf{B}} \leftarrow sgn(\mathbf{S}(\mathbf{W})\mathbf{E})$
6:     Match decoded $\tilde{\mathbf{B}}$ with target watermark $\mathbf{B}$
7:     Compute the watermark detection rate $\eta_F = \mathcal{V}_W(\mathbf{W}, \mathbf{B}, \boldsymbol{\theta})$
8:     Compute p-value corresponding to the detection rate $\eta_T$ and $\eta_F$
Output: p-value corresponding to the detection rate $\eta_T$ and $\eta_F$.
```

6 Experimental Results

This section illustrates the empirical study of the proposed FedIPR in terms of *fidelity*, *significance* and *robustness* of watermarks. Superior detection performances of both backdoor-based watermarks and feature-based watermarks in the presence of *Challenge A and B* demonstrate that FedIPR

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 3
Reported Experiment Results Under Different Settings for Proposed Feature-Based Watermarks and Backdoor-Based Watermarks

Architecture	Datasets	Watermarks		Metrics			Freeriders
		Feature-based	Backdoor-based	Fidelity	Significance	Robustness	
AlexNet	CIFAR10	Fig. 8, 9	Table 6	Fig. 7	Table 5, 7	Fig. 10, 11, 12. Table 8, 9, 10	Fig. 14
ResNet18	CIFAR100	Fig. 8	Table 6	Fig. 7	Table 5, 7	Fig. 10, 11, 12. Table 8, 9	Fig. 14
DistlBERT	SST2, QNLI	Fig. 8		Fig. 7	Table 5		

provides a reliable and robust scheme for FedDNN ownership verification.

6.1 Experiment Settings

This subsection illustrates the settings of the empirical study of our FedIPR framework, which is summarized in Table 3.

DNN Model Architectures. The deep neural network architectures we investigated include the well-known AlexNet, ResNet-18, and DistilBERT[46]. For convolution neural networks, feature-based binary watermarks are embedded into normalization scale weights $\mathbf{W}_\gamma$ of multiple convolution layers in AlexNet and ResNet-18; for tranformer-based neural networks, feature-based watermarks are embedded into *Layer-Normalization* scale weights $\mathbf{W}_\gamma$ of multiple encoders in DistilBERT. The detailed model architectures are shown in Appendix C, available in the online supplemental material.

Datasets. For image classification tasks, FedIPR is evaluated on CIFAR10 and CIFAR100 datasets, and for natural language processing tasks, FedIPR is evaluated on GLUE benchmark including SST2 and QNLI datasets.

Federated Learning Settings. We simulate a horizontal federated learning setting in which clients upload local models in each communication round, and the server adopts `Fedavg`[13] algorithm to aggregate the local models. Detailed experimental hyper-parameters to conduct FedIPR are listed in Appendix C, available in the online supplemental material. Our source codes for implementation are available at https://github.com/purp1eHaze/FedIPR.

添加watermark会对模型性能和稳定性造成明显影响。

6.2 Evaluation Metrics

Following previous DNN watermarking methods[5], [7], [12], to measure the *fidelity*, *watermark significance* and *robustness* of the proposed FedIPR framework, we apply a set of metrics as below:

Fidelity. We use classification accuracy on the main task Acc_{main} as the metrics for *fidelity*. It is expected classification accuracy should not be degraded by watermarks embedded in FedDNN (see Section 6.3 for experimental results).

Watermark Significance. The *watermark significance* measures the statistical significance that the watermarks can provide to rightfully support the ownership verification. We treat the watermark detection as a hypothesis testing process (illustrated in Algorithm 4), the watermark significance is calculated in two phases:

- *Watermark Detection Rate.* In the first phase, the watermark detection could be formulated as a classification problem which returns the watermark detection rate η_T and η_F (defined in Section 4.1).
- *Statistical Significance (p-Value).* In the second phase, we further adopt the p-value of hypothesis testing to quantify the statistical significance of watermarks.

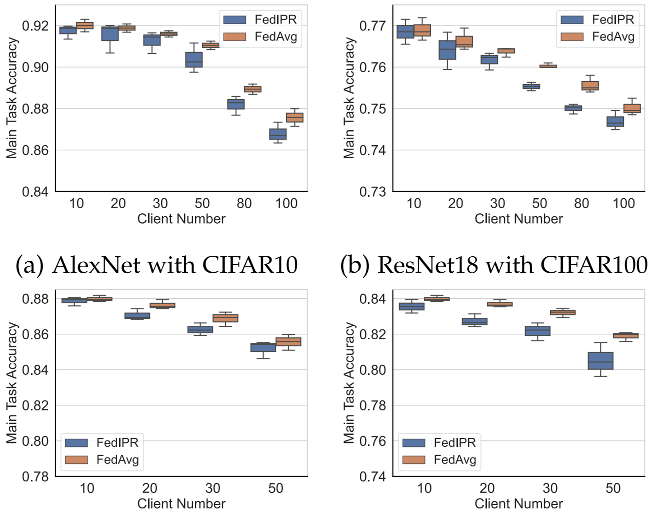
Robustness. We measure the detection rate and statistical significance of watermarks with/without training strategies and attacks to report the robustness.

6.3 Fidelity

We compare the main task performance Acc_{main} of FedIPR against `FedAvg` to report the fidelity of the proposed FedIPR. In four different training tasks, varying number (from 10 to 100) of clients may decide to embed different bit-length of backdoor-based watermarks (20 to 100 per client) and feature-based watermarks (50 to 500 bits per client).

Fig. 7a, 7b, 7c, and 7d present the worst drop of classification accuracy Acc_{main}. It is observed that under various watermarking settings, slight model performance drop (not more than 2% as compared with that of `Fedavg`) is observed for four seperated tasks.

Table 4 reports the main task accuracy with backdoor-based watermarking and feature-based watermarking respectively, the results show that the model performance drop (not more than 2 percent) is mainly caused by feature-based watermarking, a possible reason is that the regularization of feature-based watermarks may lead model parameters $\mathbf{W}$ to converge in a subspace of total space.



(a) AlexNet with CIFAR10

(b) ResNet18 with CIFAR100

(c) DistilBERT with SST2

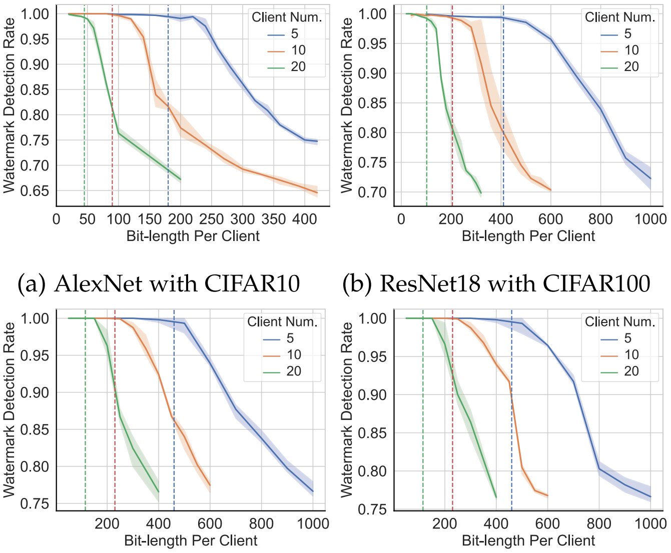
(d) DistilBERT with QNLI

Fig. 7. Figure (a)-(d), respectively, illustrate the main task accuracy Acc_{main} in image and text classification tasks with varying number K of total clients (from 10 to 100), the results are based on cases of varying settings of feature-based and backdoor-based watermarks, the main task accuracy Acc_{main} of FedIPR has slight dropped (not more than 2%) compared to `FedAvg` scheme.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 4
In the FedIPR Setting With 20 Clients, Table Shows the Main Task Accuracy Acc_{main} With Different Watermarking Methods, the CIFAR10 and CIFAR100 Datasets are Correspondly Trained With AlexNet and ResNet

Dataset	Backdoor-based		Feature-based		Bassline
	$N_{\mathbf{T}} = 50$	$N_{\mathbf{T}} = 100$	$N = 50$	$N = 100$	
CIFAR10	91.69% ± 0.15%	91.53% ± 0.18%	90.89% ± 0.23%	90.62% ± 0.29%	91.72% ± 0.12%
CIFAR100	76.47% ± 0.26%	76.32% ± 0.13%	75.12% ± 0.27%	74.29% ± 0.33%	76.52% ± 0.23%



(a) AlexNet with CIFAR10 (b) ResNet18 with CIFAR100

(c) DistilBERT with SST2 (d) DistilBERT with QNLI

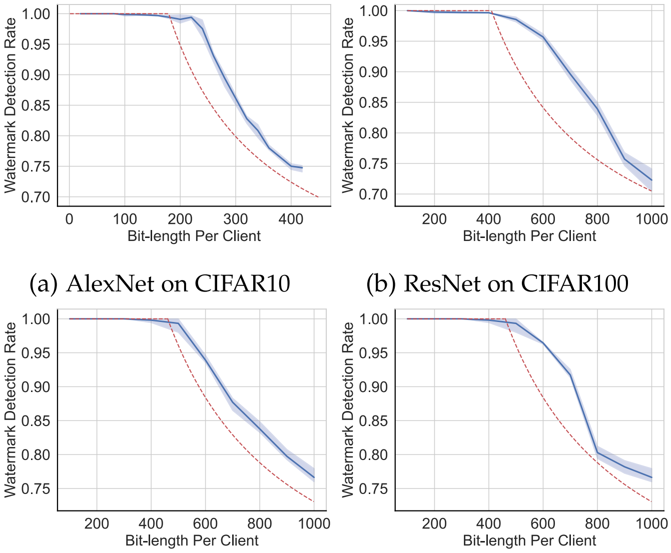
Fig. 8. Figure (a)-(d), respectively, illustrate the feature-based watermark detection rate η_F in image and text classification tasks with varying bit-length per client, in SFL with K = 5, 10, 20 clients, the dot vertical line indicates M/K, which is the theoretical bound given by Theorem 1.

6.4 Watermark Significance

We present the watermark detection rate and statistical significance to report the watermark significance of the proposed FedIPR framework.

Feature-Based Watermarks. Fig. 8a, 8b, 8c, and 8d illustrate feature-based watermark detection rates η_F of varying bit-length of feature-based watermarks, respectively on four different datasets. For convenience, we take that each client in SFL embeds the same length N of watermarks, the significance of feature-based watermarks is as below:

- *Case 1:* As shown in Fig. 8, the detection rate η_F remains constant (100%) within the vertical line (i.e., M/K), where the total bit-length KN assigned by multiple (K = 5, 10 or 20) clients does not exceed the capacity of network parameters, which is decided by the channel number M of parameter $\mathbf{S}(\mathbf{W}) = \mathbf{W}_\gamma$, respectively, e.g., $M = 896$ channels across the last 3 layers for AlexNet and $M = 2048$ channels across the last 4 layers for ResNet18 (illustrated in Appendix C, available in the online supplemental material). Therefore, when the total bit-length KN assigned by clients does not exceed the channel number M, almost all bits of feature-based watermarks can be reliably detected, which is in accordance to the Case 1 of Theorem 1[7].
- *Case 2:* When total length of watermarks KN exceeds the channel number M ($KN > M$), Fig. 9 presents the detection rate η_F drops to about 80% due to the conflicts of overlapping watermark assignments, yet the measured η_F is greater than the lower bound given by Case 2 of Theorem 1 (denoted by the red dot line).



(a) AlexNet on CIFAR10 (b) ResNet on CIFAR100

(c) DistilBERT on SST2 (d) DistilBERT on QNLI

Fig. 9. Figure provides the lower bound (red dot line) of feature-based watermark detection rate η_F given by Theorem 1, and the empirical results (blue line) are demonstrated to be above the theoretical bound (Case 2) in a SFL setting of $K = 5$ clients.

TABLE 5
In the Worst Case of Detection Rate, Table Shows the Statistical Significance of Feature-Based Watermarks

Task	CIFAR10	CIFAR100	SST2	QNLI
N Per Client	400	400	1000	1000
Detection Rate η_F	75%	71%	76%	75%
p-value	1.29e-24	1.14e-17	8.61e-64	6.73e-59

As illustrated in *Case 2*, feature-based watermarks embedded by $K = 5$ clients in SFL may conflict with each other. We give some examples of statistical significance by p-value in Table 5, even in the worst case of experiments on four different tasks, the p-value of watermarks is guaranteed below 1.17e-17, which provides a strong evidence to support claim of ownership.

According to Case 1 of Theorem 1, when $KN < M$, feature-based watermarks can be effectively embedded. In order to meet the confidence requirement, each client needs

[7] If $KN \leq M$, then there exists $\mathbf{W}$ such that $\eta_F = 1$, the results shown that η_F sometimes goes slightly below the lower bound (100%) provided by Case 1 of Theorem 1. We believe it is because the training is a multi-task optimization process as defined in Eq. (6), watermarking optimization $L_{\mathbf{B},\theta}$ is affected by the main training task optimization L_D, so the solution of watermarking is compromised to maintain the main task performance.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 6
Table Presents the Superior Backdoor-Based Watermark Detection Rate (Above 95%)

Model/Dataset	Client Num.	Trigger sample number $N_{\mathbf{T}}$ per client					
		50	100	150	200	250	300
AlexNet/CIFAR10	20	99.34% ± 0.31%	99.30% ± 0.60%	99.35% ± 0.31%	99.03% ± 0.57%	99.17% ± 0.47%	98.85% ± 0.69%
	10	99.59% ± 0.23%	98.92 % ± 0.20%	98.45% ± 0.67%	98.24% ± 0.57%	98.43% ± 0.15 %	97.56% ± 1.07%
	5	99.29% ± 0.38%	99.03% ± 0.44%	98.15% ± 0.74%	98.71% ± 0.43%	98.28% ± 0.30%	98.39% ± 0.64%
ResNet18/CIFAR100	20	99.64% ± 0.31%	99.60% ± 0.20%	99.35% ± 0.31%	99.59% ± 0.46%	99.93% ± 0.05%	99.92% ± 0.07%
	10	99.86% ± 0.05%	99.58% ± 0.41%	98.56% ± 0.57%	99.84% ± 0.04%	99.83% ± 0.15 %	99.88% ± 0.03%
	5	98.89% ± 0.80%	98.54% ± 1.3%	99.07% ± 2.34%	98.94% ± 0.73%	99.45% ± 0.06%	98.44% ± 0.25%

Respectively, table illustrates η_T of varying bit-length $N_{\mathbf{T}}$ of watermarks, where the datasets investigated include CIFAR10 and CIFAR100 datasets and the client number is 5, 10, 20.

a bit-length N larger than 40, when the number of clients scales to 10^4, a large number of channels M is required, i.e., a large model is required. In industrial practice, as the number of clients involved increases, the data and the model become larger[47]. It is a challenging task to implement FedIPR at scale, given the huge computation costs, we will solve this challenge in our future work.

Backdoor-Based Watermarks. Table 6 illustrates the detection rate η_T[8] and statistical significance of backdoor-based watermarks, where different number of clients (K = 5, 10 or 20) embed backdoor-based watermarks (triggers) generated by Projected Gradient Descent (PGD) method [48]. The results show that the watermark detection rate η_T almost keeps constant even the trigger number per client increases as much as $N_{\mathbf{T}} = 300$. Moreover, detection rate η_T of watermarks embedded in the more complex ResNet18 is as stable as those watermarks embedded in AlexNet. Also, it is noticed that the detection rate is not influenced by the varying number $N_{\mathbf{T}}$ of backdoor samples. We ascribe the stable detection rate η_T to the generalization capability of over-parameterized networks as demonstrated in [49], [50].

While with a large set of backdoor-based watermarks are embedded in FedDNN model, we give some examples of the statistical significance by p-value in Table 7. Even if the detection rate is lower than 100%, the p-value of watermarks is guaranteed below 4.02e-142, which provides a strong evidence to support claim of ownership.

6.5 Robustness under Federated Learning Strategies

As illustrated in technical *Challenge B* of Section 4.3, strategies like differential privacy[42], client selection[13] and defensive aggregation mechanism[22], [23], [24] are widely used for privacy, security and efficiency in secure federated learning. Those strategies intrinsically bring performance decades on the main classification task. Respectively, we evaluate the detection rate η_F and η_T of watermarks under *Challenge B* to report the robustness of FedIPR.

6.5.1 Robustness Against Differential Privacy

We adopt the Gaussian noise-based method to provide differential privacy guarantee for federated learning. Specifically, we vary the standard deviation σ of Gaussian noise on the local models before clients send local models to the server. As Fig. 10a and 10b show, the main task performance Acc_{main} decreases severely as the σ of noise increases, and the feature-based detection rate η_F and backdoor-based detection rate η_T drop a little while the Acc_{main} is within usable range (more than 85%). In a concrete way, when σ equals 0.003, classification accuracy Acc_{main}, detection rate η_F and η_T keep a high performance, which demonstrates the robustness of watermarks under differential privacy strategy.

We provide some examples of statistical significance by p-value in Table 8, even in the worst case of detection rate, the p-value of watermarks is guaranteed below 2.89e-15, which provides a strong evidence to support claim of ownership.

6.5.2 Robustness Against Client Selection

We select cK of K clients ($c < 1$) to participate training in each epoch for communication efficiency. Fig. 11 shows that the watermarks could not be removed even the sample ratio c is as low as 0.25. More specifically, when the sample ratio is larger than 0.2, the main classification accuracy Acc_{main} and detection rate η_T and η_F keep constant. This result gives a lower bound of client sampling rate in which watermarks can be effectively embedded and verified.

We give some examples of statistical significance by p-value in Table 9, even in the worst case of detection rate, the p-value of watermarks is guaranteed below 7.02e-20, which provides a strong evidence to support claim of ownership.

6.5.3 Robustness Against Defensive Aggregation

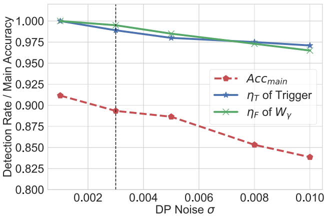
Our experiments show that even when defensive methods like Trimmed-Mean, Krum, Bulyan[22], [23], [24] are employed to defense byzantine attacks, backdoor-based watermark detection rates of more than 63.25% can still be maintained, which means a near 100 % probability of detected

TABLE 7
In the Worst Cases of Detection Rate, Table Shows Statistical Significance of Backdoor-Based Watermarks

Task	CIFAR10	CIFAR10	CIFAR100	CIFAR100
Client Number K	10	5	10	5
$N_{\mathbf{T}}$ Per Client	300	150	150	100
Detection Rate η_T	97%	98%	98%	98%
p-value	1.86e-275	4.02e-142	5.35e-289	4.85e-193

[8] The trigger samples are regarded as correctly detected when the designated targeted adversarial labels are returned.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

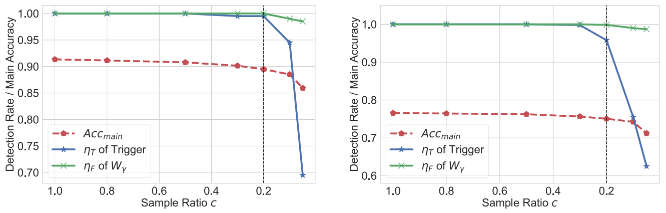


(a) DP Noise σ with AlexNet (b) DP Noise σ with ResNet

Fig. 10. This figure describes performance of FedIPR under differential privacy strategies using random noise to protect exchanged model information. In a federated learning setting of 10 clients, respectively, figure (a)-(b) illustrate feature-based detection rate η_F and backdoor-based detection rate η_T under varying differential private noise σ, where the dot lines illustrate the main task accuracy Acc_{main}.

TABLE 8
In the Worst Case of Detection Rate, Table Shows the Statistical Significance of Watermarks Under Differential Privacy Strategy Using Random Noise

Task	CIFAR10	CIFAR10	CIFAR100	CIFAR100
Watermark Type	Feature	Backdoor	Feature	Backdoor
$N/N_{\mathbf{T}}$ Per Client	80	80	80	80
Detection Rate	96.25%	97.50%	91.25%	93.75%
p-value	7.06e-20	2.56e-75	2.89e-15	2.28e-143



(a) Sample Ratio with AlexNet (b) Sample Ratio with ResNet

Fig. 11. Figure describes the robustness of FedIPR under client selection strategy. In a federated learning setting of 10 clients, respectively, figure (a)-(b) illustrate feature-based detection rate η_F and backdoor-based detection rate η_T under different sample ratio c, whereas the dot lines illustrate the main task accuracy Acc_{main}.

TABLE 9
In the Worst Case of Detection Rate, Table Shows Statistical Significance of Watermarks Under Client Selection Strategy.

Task	CIFAR10	CIFAR10	CIFAR100	CIFAR100
Watermark Type	Feature	Backdoor	Feature	Backdoor
$N/N_{\mathbf{T}}$ Per Client	80	80	80	80
Detection Rate	97.50%	68.75%	96.25%	62.50%
p-value	2.68e-21	2.74e-36	7.02e-20	6.60e-79

plagiarism is guaranteed with p-value less than $1e^{-30}$. The result is shown in Table 10.

While for feature-based watermarks under defensive methods in SFL, as presented in Table 11, the detection rate remains above 97%, the p-value of watermarks is guaranteed below 2.68e-21, which provides a strong evidence to support claim of ownership.

6.6 Robustness against Removal Attack

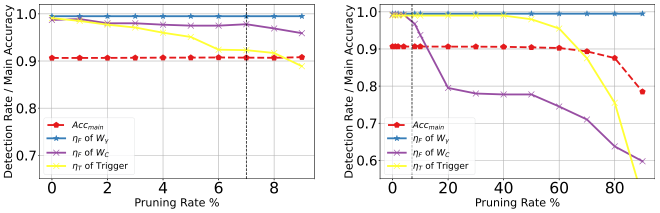
In this subsection, we showcase that FedIPR are robust against removal attacks conducted by plagiarizers that attempt to remove the watermarks. The feature-based watermarks embedded in normalization layer are shown to be especially persistent against both fine-tuning attack and pruning attack, while those watermarks in the convolution layers are not.

TABLE 10
Statistical Significance of Backdoor-Based Watermarks Under Defensive Aggregation, Where 10 Clients Train AlexNet With CIFAR10 Dataset

Method	Bulyan	Multi-Krum	Trim-mean	FedAvg
$N_{\mathbf{T}}$ Per Client	80	80	80	80
Detection Rate η_T	68.67%	79.82%	63.25%	98.82%
p-value	5.24e-35	1.74e-47	4.02e-30	7.20e-78

TABLE 11
Statistical Significance of Feature-Based Watermarks Under Defensive Aggregation, Where 10 Clients Train AlexNet With CIFAR10 Dataset

Method	Bulyan	Multi-Krum	Trim-mean	FedAvg
N Per Client	80	80	80	80
Detection Rate η_F	98.75%	100%	97.5%	100%
p-value	6.61e-23	0	2.68e-21	0



(a) Fine-tuning Epoch/5 (b) Pruning Rate

Fig. 12. Figure describes the robustness of our FedIPR under removal attacks. In a federated learning setting, K = 10 clients train AlexNet with CIFAR10 dataset. The dot lines in figure (a) and (b) illustrate the main task classification accuracy Acc_{main} under diverse settings. Respectively, figure (a) illustrates feature-based and backdoor-based watermark detection rate against model finetuning attack in 50 epochs; figure (b) illustrates feature-based and backdoor-based watermark detection rate against model pruning attack with varying pruning rate. Note that detection rates η_F of feature-based watermarks in convolution layer are severely degraded, but detection rates η_F in both normalization layer ($\mathbf{W}_\gamma$) are persistent.

6.6.1 Robustness Against Fine-Tuning Attack

Fine-tuning attack on watermarks is conducted to train the network without the presence of the regularization term, *i.e.*, $L_{\mathbf{T}}$ and $L_{\mathbf{B},\theta}$. In Fig. 12a, it is observed that the detection rate η_F of watermarks embedded with normalization layer ($\mathbf{W}_\gamma$) remains at 100% (blue curve). In contrast, the detection rate η_F of watermarks embedded with convolution layer ($\mathbf{W}_C$) drops significantly (purple curve). The superior robustness of feature-based watermarks embedded in normalization layer is in accordance to the observations reported in [36]. While for backdoor-based watermarks, the detection rate η_T remains above 90% (yellow curve), which is also robust against fine-tuning attack.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 12
Table Illustrates the Watermark Detection Rate η_T, η_F and Main Accuracy Acc_{main} in a Non-IID Federated Learning Setting, of Image Classification Tasks Including AlexNet on CIFAR10 Dataset and ResNet on CIFAR100 Dataset

Resnet	$\beta = 0.1$						
	N Per Client	50	100	200	300	400	500
	Accuracy Acc_{main}	68.61% ± 0.14%	68.54% ± 0.20%	68.33% ± 0.05%	67.55% ±0.07 %	67.38% ±0.09 %	67.33% ± 0.10%
	Backdoor-based η_T	99.73% ± 0.14%	99.74% ± 0.24%	99.85% ± 0.13%	99.75% ± 0.23%	99.73% ± 0.19%	99.69% ± 0.13%
	Feature-based η_F	100.0% ± 0%	99.73% ± 0.25%	99.90% ± 0.05%	98.96% ± 0.76%	81.83% ± 1.38%	78.45% ± 0.68%
	$\beta = 1.0$						
	N Per Client	50	100	200	300	400	500
	Accuracy Acc_{main}	74.52% ± 0.23%	74.48% ± 0.33%	74.64% ± 0.05 %	73.96% ± 0.35%	73.81% ± 0.13%	73.31% ± 0.06%
	Backdoor-based η_T	99.85% ± 0.14%	99.85% ± 0.15%	99.74 % ± 0.24%	99.75% ± 0.23%	0.9974 ± 0. 23%	0.9973 ± 0.13%
	Feature-based η_F	100% ± 0%	99.60% ± 0.10%	99.55% ± 0.05%	97.66% ± 0.73%	80.87% ± 1.53%	77.97% ± 0.53%
Alexnet	$\beta = 0.1$						
	N Per Client	50	100	150	200	250	300
	Accuracy Acc_{main}	82.30% ± 0.31%	82.52% ± 0.60%	82.65% ± 0.19 %	82.42% ± 1.02 %	81.49% ± 0.85 %	81.38% ± 0.93 %
	Backdoor-based η_T	99.24% ± 0.25%	99.30% ± 0.21%	99.53% ± 0.15%	99.82% ± 0.13%	99.81% ± 0.09%	99.82% ± 0.13%
	Feature-based η_F	100% +0%	99.84% ± 0.08%	93.49% ± 0.40%	86.87% ± 0.31%	79.87% ± 0.51%	76.97% ± 0.43%
	$\beta = 1.0$						
	N Per Client	50	100	150	200	250	300
	Accuracy Acc_{main}	89.76% ± 0.09%	89.50% ± 0.07%	89.46% ± 0.21%	88.53% ± 0.23%	88.60% ± 0.13%	88.43% ± 0.34%
	Backdoor-based η_T	99.25% ± 0.25%	99.34% ± 0.46%	99.54% ± 0. 23%	99.85% ± 0.15%	99.75% ± 0.31%	99.75% ± 0.31%
	Feature-based η_F	99.80% ± 0.20%	99.61% ± 0.13%	94.24% ± 0.13%	87.14% ± 0.38%	77.14% ± 0.48%	75.36% ± 0.68%

$K = 10$ clients embed feature-based and backdoor-based watermarks, with varying feature-based watermark length N per client, the trigger number is set to 80. The results including non-iid settings sampled from dirichlet distribution with $\beta = 0.1$ and 1.

6.6.2 Robustness against Pruning Attack

The pruning attack removes redundant parameters from the trained model. We evaluate the main task performance Acc_{main} and watermark detection rate η_F and η_T under pruning attack with varying pruning rates. Fig. 12b shows watermark detection rate η_F and η_T while varying proportions of network parameters are pruned. It is observed that the detection rate η_F of watermarks embedded in the normalization layer is stable all the time, while η_F with $\mathbf{W}_C$ are severely degraded. This fact shows that the watermarks on normalization parameters are more robust against pruning attack. While for backdoor-based watermarks, the detection rate η_T remains above 90% (yellow curve) when pruning rate is less than 60%, which is also robust against pruning attack.

那是因为pruning不涉及norm layer吧？

6.7 Robustness of Triggers against Adversary

An adversary might try to obtain counterfeited triggers $\hat{\mathbf{T}}$ by generating adversarial examples with a surrogate network $\mathbb{N}_{sur}$, and attempt to pass the ownership verification of target model $\mathbb{N}$ with those triggers $\hat{\mathbf{T}}$ as follows:

$$\mathcal{V}_B(\mathbb{N}, \hat{\mathbf{T}}) = \begin{cases} \text{TRUE}, & \text{if } \mathbb{E}_{\hat{\mathbf{T}}}(\mathbb{I}(\mathbf{Y}_{\hat{\mathbf{T}}} \neq \mathbb{N}(\mathbf{X}_{\hat{\mathbf{T}}}))) \leq \epsilon_B, \\ \text{FALSE}, & \text{otherwise}, \end{cases} \tag{21}$$

We give results in 2 cases to show that in SFL setting, adversary has little knowledge of the original training data or backdoor triggers, thus it is hard for an adversary to obtain triggers that can pass the verification.

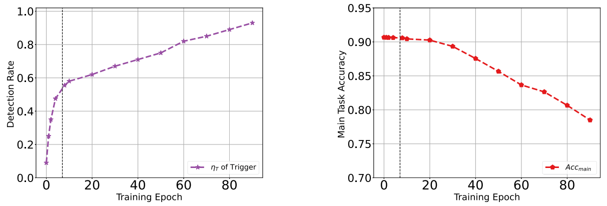
Case1. If the attacker randomly generates some adversarial samples from surrogate model $\mathbb{N}_{sur}$ as triggers $\hat{\mathbf{T}}$, but does not retrain the model with the triggers $\hat{\mathbf{T}}$. When the triggers $\hat{\mathbf{T}}$ are input to the API for verification, the accuracy that the model outputs the target label is an almost random guessing (e.g., in a CIFAR10 classification task, the detection rate of trigger is about 10%), which is shown in the following Table 13. The results indicates that the random generated triggers can not pass the verification without retraining.

Case2. If the attacker randomly generates some triggers $\hat{\mathbf{T}}$ and retrains the model $\mathbb{N}$ with those triggers, results in the following Fig. 13 show that the 200 trigger images can be embedded with 80 epochs of training, the triggers are generated with unrelated base images with CIFAR10 dataset, which will result in main task accuracy decades larger than

TABLE 13
In the CIFAR10 Classification Task With AlexNet, an Attacker Randomly Generates Some Adversarial Samples $\hat{\mathbf{T}}$ From the Surrogate Model $\mathbb{N}_{sur}$ With a Set of Base Images (Unrelated With the Private Training Data)

Trigger type	Number of Triggers				
	100	200	300	400	500
True trigger $\mathbf{T}$	100%	99.63%	99.65%	99.52%	99.72%
Counterfeited $\hat{\mathbf{T}}$	9.34%	11.21%	10.15%	9.36%	11.09%

Table shows the accuracy that the model $\mathbb{N}$ outputs triggers as targeted label.



(a) Trigger Detection Rate (b) Main Task Accuracy

Fig. 13. Figure presents the trigger detection accuracy and the main accuracy while adversary retrains the model $\mathbb{N}$ with triggers $\hat{\mathbf{T}}$, in the CIFAR10 classification task with AlexNet.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

TABLE 14
In the Worst Case of Detection Rate, Table Shows the Statistical Significance of Watermarks Under Non-IID SFL

Task	CIFAR10	CIFAR10	CIFAR100	CIFAR100
Non-iid β	0.1	1	0.1	1
N Per Client	300	300	500	500
Detection Rate η_F	76%	75%	78%	77%
p-value	2.42e-20	7.16e-19	4.76e-38	2.33e-35

10 percent. Such performance descend defeats the purpose of plagiarism to obtain an existing model at no cost.

6.8 Robustness Under Non-IID Setting

In federated learning, data distributions across clients are often not identical and independent distributed (iid). We also evaluate FedIPR in *lable-skew* non-iid federated learning setting, where we assume each client's training examples are drawn with class labels following a *dirichlet distribution* [51], $\beta > 0$ is the concentration parameter controlling the identicalness among users.

In our experiments, we conduct image classification experiments with AlexNet on CIFAR10 and ResNet on CIFAR100, the dirichlet parameter is $\beta = 0.1$ and 1. The results presented in Table 12 indicate that FedIPR works with non-iid setting.

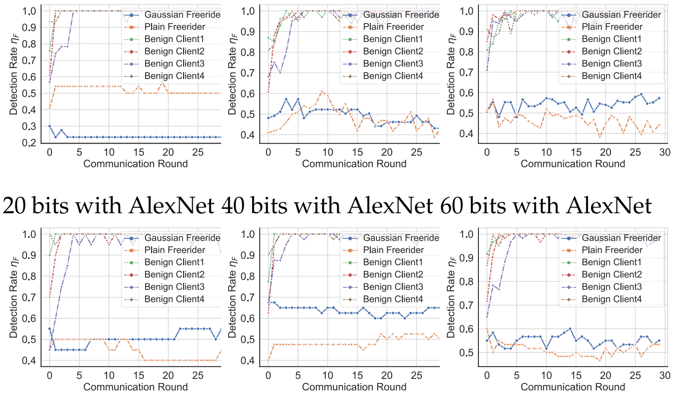
In non-iid setting, while a large set of feature-based watermarks are embedded in FedDNN model, we give some examples of statistical significance by p-value in the worst case of detection rate in Table 14.

As shown in Table 14, even in the worst case, the p-value of watermarks can be guaranteed below 7.16e-19, which provides a strong evidence to support claim of ownership.

6.9 Watermarks Defeat Freerider Attacks

As a precaution method for freerider attack, watermarks are embedded into the FedDNN, the benign clients can verify ownership by extracting predefined watermarks from FedDNN, while freeriders can not detect watermarks because they do not perform actual training. We conduct experiments testing the local models by three types of clients including plain freerider, freerider with Gaussian noise (defined in Section 3.2) and benign clients that contribute data and computation.

We consider a setting that the server conducts feature-based verification on each client's local models, in a 22-clients federated learning including one freerider client with Gaussian noise and one plain freerider client with previous local models. In each communication round, the results are presented in Fig. 14, which show that the server can detect the benign clients' watermarks in the global model at quite early stage (in 30 communication rounds) of FedDNN model training, the watermark detection rate η_F is nearly 100%; while the freeriders failed to verify their watermarks because they do not contribute actual training, the η_F detection rate is an almost random guess (50%).



20 bits with AlexNet 40 bits with AlexNet 60 bits with AlexNet

20 bits with ResNet 40 bits with ResNet 60 bits with ResNet

Fig. 14. Comparisons between three different types of clients including: (1) freerider clients with previous local models. (orange lines) (2) freerider clients disguised with Gaussian noise. (blue lines) (3) four benign clients in a 20 client SFL. The feature-based watermark detection rate η_F is measured in each communication round. Note the sharp contrast of η_F between four benign clients and two freerider clients (plotted in orange and blue).

7 Conclusion

This paper presents a novel ownership verification scheme to protect the Intellectual Property Right (IPR) of Federated DNN models against external plagiarizers who illegally copy, re-distribute the models. To our best knowledge, it is the first ownership verification scheme that aims to protect model intellectual property rights under secure federated learning setting. This work addresses a crucial issue remained open in secure federated learning research, since the protection of valuable federated learning models is as important as protecting data privacy.

On the technical side, this work demonstrates that reliable and persistent watermarks could be embedded into local models without disclosing the presence and extraction parameters of these watermarks. In particular, normalization scale parameters based on watermarks are extremely robust under federated learning strategies and against removal attacks. We wish that the formulation illustrated in this paper will lead to watermark embedding and verification in various federated learning settings.

Acknowledgments

The work was done when Bowen Li was an intern at WeBank.

References

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[2] C. Wang et al., "Voxpopuli: A large-scale multilingual speech corpus for representation learning, semi-supervised learning and interpretation," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 993–1003.

[3] C. Chelba et al., "One billion word benchmark for measuring progress in statistical language modeling," 2013, *arXiv:1312.3005*.

[4] Y. Zhu et al., "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 19–27.

[5] L. Fan, K. W. Ng, C. S. Chan, and Q. Yang, "DeepIP: Deep neural network intellectual property protection with passports," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jun. 14, 2021, doi: 10.1109/TPAMI.2021.3088846.

[6] D. S. Ong, C. S. Chan, K. W. Ng, L. Fan, and Q. Yang, "Protecting intellectual property of generative adversarial networks from ambiguity attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 3629–3638.

[7] J. Zhang et al., "Deep model intellectual property protection via deep watermarking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 8, pp. 4005–4020, Aug. 2021.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

[8] J. H. Lim, C. S. Chan, K. W. Ng, L. Fan, and Q. Yang, "Protect, show, attend and tell: Empowering image captioning models with ownership protection," *Pattern Recognit.*, vol. 122, 2022, Art. no. 108285.

[9] Y. Uchida, Y. Nagai, S. Sakazawa, and S. Satoh, "Embedding watermarks into deep neural networks," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2017, pp. 269–277.

[10] H. Chen, B. Darvish Rohani, and F. Koushanfar, "DeepMarks: A Digital Fingerprinting Framework for Deep Neural Networks," Apr. 2018, *arXiv: 1804.03648*.

[11] B. Darvish Rouhani, H. Chen, and F. Koushanfar, "DeepSigns: A Generic Watermarking Framework for IP Protection of Deep Learning Models," Apr. 2018, *arXiv: 1804.00750*.

[12] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *Proc. 27th USENIX Secur. Symp.*, 2018, pp. 1615–1631.

[13] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artif. Intell. Statist.*, 2017, pp. 1273–1282. [Online]. Available: http://proceedings.mlr.press/v54/mcmahan17a.html

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

[15] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends® Mach. Learn.*, 2019. [Online]. Available: http://arxiv.org/abs/1912.04977

[16] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 14 747–14 756. [Online]. Available: http://papers.nips.cc/paper/9617-deep-leakage-from-gradients

[17] X. Luo, Y. Wu, X. Xiao, and B. C. Ooi, "Feature inference attack on model predictions in vertical federated learning," in *Proc. IEEE 37th Int. Conf. Data Eng.*, 2021, pp. 181–192.

[18] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 5, pp. 1333–1345, May 2018.

[19] M. Abadi et al., "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 308–318.

[20] T. Ryffel, D. Pointcheval, and F. R. Bach, "ARIANN: low-interaction privacy-preserving deep learning via function secret sharing," *CoRR*, 2020. [Online]. Available: https://arxiv.org/abs/2006.04593

[21] X. Zhang, H. Gu, L. Fan, K. Chen, and Q. Yang, "No free lunch theorem for security and utility in federated learning," 2022, *arXiv:2203.05816*.

[22] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. 31st Int. Conf. Neural Informat. Process. Syst.*, 2017, pp. 118–128.

[23] R. Guerraoui, S. Rouault et al., "The hidden vulnerability of distributed learning in byzantium," in *Proc. Int. Conf. MachineMach. Learn.*, 2018, pp. 3521–3530.

[24] D. Yin, Y. Chen, R. Kannan, and P. Bartlett, "Byzantine-robust distributed learning: Towards optimal statistical rates," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5650–5659.

[25] Y. Fraboni, R. Vidal, and M. Lorenzi, "Free-rider attacks on model aggregation in federated learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2021, pp. 1846–1854.

[26] J. Lin, M. Du, and J. Liu, "Free-riders in federated learning: Attacks and defenses," 2019, *arXiv:1911.12560*.

[27] L. Lyu, H. Yu, and Q. Yang, "Threats to federated learning: A survey," 2020, *arXiv:2003.02133*.

[28] F. Boenisch, "A survey on model watermarking neural networks," 2020, *arXiv:2009.12153*.

[29] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, 2015, pp. 1310–1321.

[30] M. de Boer, Ai as a target and tool: An attacker's perspective on ML, 2020. [Online]. Available: https://www.gartner.com/en/documents/3939991

[31] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction {APIs}," in *Proc. 25th USENIX Secur. Symp.*, 2016, pp. 601–618.

[32] T. Orekondy, B. Schiele, and M. Fritz, "Knockoff nets: Stealing functionality of black-box models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4954–4963.

[33] J. Zhang et al., "Protecting intellectual property of deep neural networks with watermarking," in *Proc. Asia Conf. Comput. Commun. Secur.*, 2018, pp. 159–172.

[34] N. Lukas, Y. Zhang, and F. Kerschbaum, "Deep neural network fingerprinting by conferrable adversarial examples," in *Proc. 9th Int. Conf. Learn. Representations*, 2021.

[35] J. Zhang, D. Chen, J. Liao, W. Zhang, G. Hua, and N. Yu, "Passport-aware normalization for deep model protection," in *Proc. Adv. Neural Informat. Process. Syst.*, 2020, pp. 22 619–22 628. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/ff1418e8cc993fe8abcfe3ce2003e5c5-Paper.pdf

[36] L. Fan, K. W. Ng, and C. S. Chan, "Rethinking deep neural network ownership verification: Embedding passports to defeat ambiguity attacks," in *Proc. Adv. Neural Informat. Process. Syst.*, 2019, pp. 4714–4723.

[37] G. Xu, H. Li, S. Liu, K. Yang, and X. Lin, "VerifyNet: Secure and verifiable federated learning," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 911–926, 2019.

[38] G. Han, T. Zhang, Y. Zhang, G. Xu, J. Sun, and J. Cao, "Verifiable and privacy preserving federated learning without fully trusted centers," *J. Ambient Intell. Humanized Comput.*, vol. 13, no. 3, pp. 1431–1441, 2022.

[39] B. G. Atli, Y. Xia, S. Marchal, and N. Asokan, "Waffle: Watermarking in federated learning," 2020, *arXiv:2008.07298*.

[40] X. Liu, S. Shao, Y. Yang, K. Wu, W. Yang, and H. Fang, "Secure federated learning model verification: A client-side backdoor triggered watermarking scheme," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, 2021, pp. 2414–2419.

[41] Z. Sun, P. Kairouz, A. Theertha Suresh, and H. B. McMahan, "Can You Really Backdoor Federated Learning?," Nov. 2019, *arXiv: 1911.07963*.

[42] K. Wei et al., "Federated learning with differential privacy: Algorithms and performance analysis," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 3454–3469, 2020.

[43] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," 2018, *arXiv:1807.00459*.

[44] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," 2017, *arXiv:1706.06083*.

[45] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Informat. Process. Syst.*, 2017, pp. 6000–6010.

[46] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," 2019, *arXiv:1910.01108*.

[47] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient federated learning via guided participant selection," in *Proc. 15th {USENIX} Symp. Operating Syst. Des. Implementation*, 2021, pp. 19–35.

[48] A. Nguyen, J. Yosinski, and J. Clune, "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 427–436.

[49] Z. Allen-Zhu, Y. Li, and Z. Song, "A convergence theory for deep learning via over-parameterization," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 242–252.

[50] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM*, vol. 64, no. 3, pp. 107–115, 2021.

[51] Q. Li, Y. Diao, Q. Chen, and B. He, "Federated learning on non-iid data silos: An experimental study," 2021, *arXiv:2102.02079*.

[52] N. Alon and K. A. Berman, "Regular hypergraphs, gordon's lemma, steinitz'lemma and invariant theory," *J. Combinatorial Theory, Ser. A*, vol. 43, no. 1, pp. 91–97, 1986.

Bowen Li received the BS degree in automation from Xi'an Jiaotong University, China, in 2019. He is currently working toward the PhD degree with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. He worked as a research intern with WeBank AI Group, WeBank, China, in 2021. His research interests include federated learning, data privacy and machine learning security.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.

Lixin Fan is the chief scientist of artificial intelligence with WeBank. His research fields include machine learning and deep learning, computer vision and pattern recognition, image and video processing, 3D big data processing, data visualization and rendering, augmented and virtual reality, mobile computing and ubiquitous computing, and intelligent man-machine interface. He is the author of more than 70 international journals and conference articles. He has worked with Nokia Research Center and Xerox Research Center Europe. He has participated in NIPS/NeurIPS, ICML, CVPR, ICCV, ECCV, IJCAI and other top artificial intelligence conferences for a long time, served as area chair of ICPR, and organized workshops in various technical fields. He is also the inventor of more than one hundred patents filed in the United States, Europe and China, and the chairman of the IEEE P2894 Explainable Artificial Intelligence (XAI) Standard Working Group.

Hanlin Gu received the BS degree in mathematics from the University of Science and Technology of China, in 2017, and the PhD degree from the Department of Mathematics, Hong Kong University of Science and Technology. He is working as a researcher with WeBank AI Group, WeBank, China in 2021. His research interests include federated learning, privacy-preserving methodology.

Jie Li received the BE degree in computer science from Zhejiang University, Hangzhou, China, in 1982, the ME degree in electronic engineering and communication systems from the China Academy of Posts and Telecommunications, Beijing, China, in 1985, and the DrEng degree from the University of ElectroCommunications, Tokyo, Japan, in 1993. He is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, where he is a professor. He was a full professor with the Department of Computer Science, University of Tsukuba, Tsukuba, Japan. He was a visiting professor with Yale University, New Haven, CT, USA; Inria Sophia Antipolis, Biot, France; and Inria Grenoble Rhône-Alpes, Montbonnot-Saint-Martin, France. His current research interests are in big data, IoT, blockchain, edge computing, OS, and modeling and performance evaluation of information systems. He is the co-chair of the IEEE Technical Community on Big Data and the IEEE Big Data Community, and the founding chair of the IEEE ComSoc Technical Committee on Big Data. He serves as an associate editor for many IEEE journals and transactions. He has also served on the program committees for several international conferences.

Qiang Yang is a fellow of Canadian Academy of Engineering (CAE) and Royal Society of Canada (RSC), Chief Artificial Intelligence Officer of WeBank, a chair professor of Computer Science and Engineering Department, Hong Kong University of Science and Technology (HKUST). He is the conference chair of AAAI-21, the honorary vice president of Chinese Association for Artificial Intelligence(CAAI), the president of Hong Kong Society of Artificial Intelligence and Robotics (HKSAIR) and the president of Investment Technology League (ITL). He is a fellow of AAAI, ACM, CAAI, IEEE, IAPR, AAAS. He was the founding editor in chief of the *ACM Transactions on Intelligent Systems and Technology (ACM TIST)* and the founding editor in chief of *IEEE Transactions on Big Data (IEEE TBD)*. He received the ACM SIGKDD Distinguished Service Award, in 2017. He had been the founding director of the Huawei's Noah's Ark Research Lab between 2012 and 2015, the founding director of HKUST's Big Data Institute, the founder of 4Paradigm and the president of IJCAI (2017-2019). His research interests are artificial intelligence, machine learning, data mining and planning.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.

Authorized licensed use limited to: National University of Singapore. Downloaded on June 05,2023 at 08:56:21 UTC from IEEE Xplore. Restrictions apply.