Journal of Systems Architecture 155 (2024) 103258

Contents lists available at ScienceDirect

Journal of Systems Architecture

journal homepage: www.elsevier.com/locate/sysarc

BR-FEEL: A backdoor resilient approach for federated edge learning with fragment-sharing

Senmao Qi [a], Hao Ma [a], Yifei Zou [a,*], Yuan Yuan [a], Peng Li [b], Dongxiao Yu [a]

[a] Institute of Intelligent Computing, School of Computer Science and Technology, Shandong University, Qingdao, 266237, PR China
[b] The School of Computer Science and Engineering, The University of Aizu, University of Aizu, Aizuwakamatsu, 965-8580, Japan

ARTICLE INFO

Keywords:
Federated edge learning
Fragment-sharing
Backdoor defense
Knowledge distillation

ABSTRACT

In the resource-constrained federated edge learning (FEEL) systems, fragment-sharing is an efficient approach for multiple clients to cooperatively train a giant model with billions of parameters. Compared with the classical federated learning schemes where the local model is fully trained and exchanged by each client, the fragment-sharing only requires each client to optionally choose a parameter-fragment to train and share, according to its storage, computing, and networking abilities. However, when the full model is no longer delivered in fragment-sharing, the backdoor attacks hidden behind the fragments become harder to be detected, which introduces formidable challenge for the security of FEEL systems. In this paper, we firstly show that the existing fragment-sharing works suffer a lot from the backdoor attacks. Then, a Backdoor-Resilient approach, named BR-FEEL, is introduced to defend against the potential backdoor attacks. Specifically, a twin model is built by each benign client to integrate the parameter-fragments from others. A knowledge distillation process is designed on each client to transfer the clean knowledge from its twin model to local model. With the twin model and knowledge distillation process, our BR-FEEL approach makes sure that the local models of the benign clients will not be backdoored. Experiments on CIFAR-10 and GTSRB datasets with MobileNetV2 and ResNet-34 are conducted. The numerical results demonstrate the efficacy of BR-FEEL on reducing attack success rates by over 90% compared to other baselines under various attack methods.

1. Introduction

With the rapid development of Artificial Intelligence and Edge Computing, the cooperative training of a large-scale model in the federated edge learning (FEEL) scenario has become a significant and interesting research topic. On one hand, the model training and sharing on the billions of parameters pose a significant challenge for resources-constrained edge systems [1,2]. On the other hand, sharing a full model to others may leak the privacy of users, since a gradient inversion attack that can recover the local data of a client from its shared model has been presented in [3]. With the above consideration, more and more researches consider the fragment-sharing as an efficient approach to cooperatively train a giant model in FEEL [4–7]. Compared with traditional methods in which the full models are exchanged, the fragment-sharing only requires the clients to optionally select a parameter-fragment to train and share, according to their storage, computing, and networks abilities. Thus, it can be recognized as a resource-friendly and privacy-preserving approach in FEEL.

Despite the above advantages, the fragment-sharing also introduces a new challenge on the security of FEEL. Specifically, a malicious client can conceal a backdoor in its fragment that is going to be shared with others [8]. When receiving and integrating the backdoor fragment, a benign client has its local model infected with the backdoor unconsciously. Then, the backdoor model of the benign client produces correct outputs for benign inputs but exhibits malicious outputs desired by the attacker when recognizing a trigger [9]. A diagram of the backdoor attack for fragment-sharing is illustrated in Fig. 1, in which the benign model has it local model infected with a backdoor and misclassify a stop sign with a black block as a pass sign, thereby introducing potential security risks.

Due to the threat of backdoor attacks on the security of FL, a series of backdoor defense strategies have been proposed, most of which are based on full model exchange and can be categorized into pre-aggregation defense, in-aggregation defense, and post-aggregation defense [9]. In pre-aggregation defense, clustering methods, e.g. Krum in [10], AFA in [11], and Auror in [12], are adopted to detect and exclude those backdoor models before model aggregation. In-aggregation defense uses robust aggregation techniques during model aggregation,

* Corresponding author.
E-mail addresses: senmao_qi@mail.sdu.edu.cn (S. Qi), haoma@mail.sdu.edu.cn (H. Ma), yfzou@sdu.edu.cn (Y. Zou), yyuan@sdu.edu.cn (Y. Yuan), pengli@u-aizu.ac.jp (P. Li), dxyu@sdu.edu.cn (D. Yu).

https://doi.org/10.1016/j.sysarc.2024.103258
Received 31 January 2024; Received in revised form 26 June 2024; Accepted 7 August 2024
Available online 9 August 2024

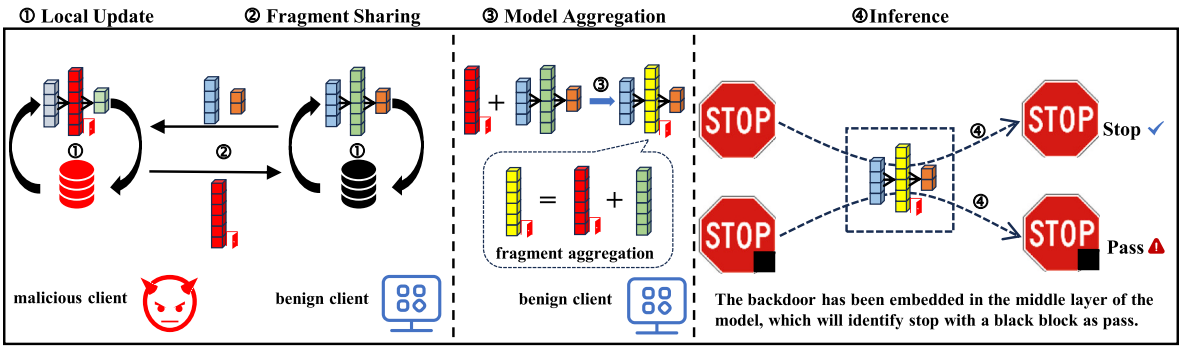
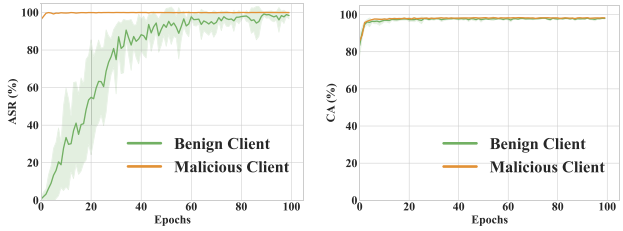


Fig. 1. A diagram of the backdoor attack in FEEL with fragment-sharing.



(a) ASR versus training epochs. (b) CA versus training epochs.

Fig. 2. Results of backdoor attack in fragment-sharing.

such as median, trimmed mean [13] or robust learning rate [14] to help the parameter server get rid of the negative impact from the malicious models. Post-aggregation defense usually rely on the individual clients to purify the aggregated global model. Techniques such as neuron pruning [15] and unlearning [16] are commonly employed for this purpose. However, all the existing studies require the full model as the background to detect, robust aggregate, or purify the backdoor models. Lacking the full model makes the previous defenses unsuitable in the fragment-sharing approach, and makes it harder to verify whether a fragment is safe or not. Consequently, a crucial question arises: *In FEEL with fragment-sharing, how can a benign client acquire useful knowledge from others without introducing potential backdoor?*

To further motivate our work, a demo based on [7] is conducted to concretely show how the backdoor attack can be conducted in FEEL with fragment-sharing. Specifically, we consider a FEEL system with four benign clients and one malicious client. Each client holds a fraction of data from the MNIST dataset [17], and locally trains a neural network according to its dataset. The neural network of each client consists of three convolutional layers and one linear layer. Due to the constrained computing and network resources, at most two layers of the neural network will be trained and shared by each benign client in each training round. The malicious client embeds a backdoor by fine-tuning on a poisoned dataset [3], introducing the backdoor into the shared model fragment. When receiving the fragments shared from others, the benign clients integrates those fragments into its local model according to the method proposed in [7]. Figs. 2(a) and 2(b) illustrate the change of attack success rate (ASR) and clean test rate (CA) with the number of training epochs. ASR reflects the probability of the model outputting the result expected by the attacker when facing input with trigger, while CA reflects the probability of the model outputting the correct result when facing clean input without trigger [9]. Notably, the ASR of the benign client steadily increases with the number of training epochs, and finally gets close to 100%. Besides, the CA of benign clients remains stable at about 97%, which means the backdoor attack cannot be detected on the clean data.

To answer the above crucial questions, this paper studies the backdoor defense in FEEL with fragment-sharing and proposes a **b**ackdoor **r**esilient **fe**derated **e**dge **l**earning algorithm (BR-FEEL for short). Specifically, we consider a decentralized FEEL scenario that includes some benign clients with clean datasets and some malicious clients with poisoned datasets. Our objective is to ensure that the benign clients can cooperatively train clean models without backdoor introduced by fragment-sharing. Considering that only the parameter-fragments are shared, the backdoor fragment cannot be detected until it is embedded in the local model. To keep its local model clean, each benign client uses a twin model to receive the fragments shared from others. Due to the existence of backdoor fragments, this twin global model is likely to exhibit backdoor behaviors. Considering that the backdoor model will not exhibit abnormal behavior when facing input without triggers [9], we adopt knowledge distillation technique and use the twin model as a teacher to guide the local model of benign clients, thereby achieving global knowledge transfer without introducing backdoor. A series of experiments are conducted to validate the efficiency and backdoor-resilience of our method. In summary, our contributions are as follows:

- To the best of our knowledge, this paper is the first to pay attention to the backdoor defense issue in FEEL with fragment-sharing. We use a demo to prove that the malicious clients in FEEL can launch backdoor attacks on the benign clients by sharing a backdoored fragment of the whole model. Our work highlights the potential risks of backdoor attacks in FEEL with fragment-sharing.
- A **b**ackdoor **r**esilient **fe**derated **e**dge **l**earning (BR-FEEL) approach is proposed to effectively help benign clients acquire knowledge from other clients without introducing backdoor. Specifically, a twin model is used by the benign clients to integrate the parameter-fragments shared by other clients. Then, the twin model can serve as a teacher network for knowledge distillation during the training of the local model. With the help of the twin model and knowledge distillation technique, our BR-FEEL approach has a strong backdoor resilience against most of the backdoor attack methods.
- We conduct extensive experiments utilizing the MobileNetV2 and ResNet-34 model on the CIFAR-10 [18] and GTSRB [19] datasets. Specifically, we introduce five common data poisoning methods, namely BadNet [20], Blend [21], Dynamic [22], Trojan [23], and Adaptive_patch [24]. The performance of different defense baselines, including Vanilla FEEL [7], Median [13], Geometric Median [25], Norm Clipping [26], and our BR-FEEL, is compared under these attack methods. Simultaneously, we explore the impact of the proportion of malicious clients and data distribution on BR-FEEL. Numerical results reveal that our BR-FEEL significantly reduces the attack success rate by more than 90% on the CIFAR-10 and GTSRB datasets compared to Vanilla FEEL, Median, and Geometric Median. Furthermore, in comparison to Norm Clipping, we achieve a reduction in the attack success rate by more than 6% and an increase in prediction accuracy by more than 40%.

Roadmap. The remainder of this paper is organized as follows. Section 2 gives the related work. Section 3 presents our FEEL model and the problem definition. The BR-FEEL are presented in Section 4. In Section 5, we conduct extensive experiments to exhibit the performance of BR-FEEL. Finally, Section 6 concludes this work.

2. Related work

In this section, we introduce related work about FEEL with fragment-sharing, summarize existing backdoor defense strategies in FL and knowledge distillation technology.

2.1. FEEL with fragment-sharing

Due to the limitations of system bandwidth [7], computing power [27], and user privacy requirements [3], there has been a growing interest in FEEL with fragment-sharing, where only a subset of the model parameters is shared among clients. In FEEL with fragment-sharing, a critical consideration is how to obtain a model fragment. Consequently, research in [28–30] has introduced optimization techniques to get a suitable model fragment, achieving a trade-off between training efficiency, communication burden, and privacy budget. Similar works can also be found in personalized federated learning [31–33], where clients share only convolutional layers and batch normalization layers to enhance local penalization. However, many existing studies can only share specific model fragments or necessitate a complex calculation process to obtain an appropriate model fragment, making them less adaptable to dynamic federated learning systems. As a solution, Wang et al. [7] propose the use of masks to derive a model fragment for communication, and present a resource-adaptive learning algorithm with theoretical convergence guarantees under arbitrary neuron assignments. This approach is viewed as a promising paradigm for FEEL with fragment-sharing. In this paper, our FEEL with fragment-sharing is built upon the framework introduced in [7].

2.2. Backdoor defense in FL

Given the potential harm of backdoor attacks in FL, numerous studies have explored backdoor defenses in FL [9,10,13,15,16]. Based on the timing of implementing defense mechanisms, backdoor defenses in FL can be categorized into Pre-AD, In-AD, and Post-AD [9]. The objective of Pre-AD defense is to filter out malicious model parameters to avoid aggregating a backdoor model. Considering the similarity among benign model parameters, clustering is introduced as an effective means to assist the parameter server in identifying similar benign model parameters. Examples include Krum [10], AFA [11], Auror [12], and FoolsGold [34], which utilize Mahalanobis distance or cosine similarity to identify such similar benign model parameters. In-AD defense aims to obtain a clean global model during the aggregation process. Techniques such as median, trimmed mean [13], and robust learning rate [14] are employed for robust model aggregation, eliminating the influence of malicious model parameters. Similarly, to minimize the impact of malicious model parameters during aggregation, differential privacy techniques are adopted, which involves normalizing local models and adding appropriate Gaussian noise to resist backdoor attacks during the aggregation process [26,35]. Post-AD defense aims to purify a backdoor global model, thus getting a clean global model. For instance, Wu et al. [15] use clean data to prune the backdoor model, removing backdoor parameters from the neural network. Similar purification methods include conducting machine unlearning on the backdoor model to render it incapable of backdoor attacks [16]. However, most existing research has primarily focused on backdoor defenses in FL with full model sharing. In FEEL with fragment-sharing, the fragment introduces additional challenges for backdoor defenses.

2.3. Knowledge distillation

Knowledge distillation, initially proposed by Hinton et al. [36], is seen as an effective method for transferring knowledge. In knowledge distillation, the student network is supervised by the knowledge provided by the teacher. For instance, the student network can be trained by imitating the output knowledge of certain layers of the teacher network [36–38]. According to the type of knowledge, knowledge distillation is divided into three categories: response-based, feature-based, and relationship-based [39]. In response-based knowledge distillation, a common approach involves having the student network learn the soft logits produced by the teacher network, which is the output result of the last fully connected layer of the neural network [36,40,41]. However, this distillation method ignores results of the middle layers of the neural network, leading to relatively limited improvement in the student network. Therefore, feature-based knowledge distillation is proposed [42–44], which uses features extracted from the middle layer of the teacher network to serve as hints for the output of the middle layer of the student model. In order to better extract teacher feature information, Li et al. [45] use supervised learning to find important features. Differing from the aforementioned distillation modes that involve learning the output results of the teacher network, relationship-based knowledge distillation provides the student network with a relationship mapping that facilitates learning from the teacher model [39]. Notable studies include the incorporation of the flow of solution procedure matrix to guide the training of the student model [46].

In recent years, knowledge distillation has also been explored in the research of backdoor defense. For example, Li et al. [47] introduce neural attention distillation to repair a backdoor model on clean dataset. Similar work is also reflected in [48,49]. In addition to the above-mentioned backdoor defense under single machine, Zhu et al. [50] deploy a generative adversarial network on the server side to generate negative samples and use adversarial distillation to repair the backdoored global model in FL. Some recent works [51,52] also point out that repairing the backdoored global model through knowledge distillation can effectively alleviate the hidden dangers of backdoor attacks in FL.

However, the existing knowledge distillation-based backdoor defense mechanisms still have some non-negligible concerns. Firstly, most works require the parameter server to have an additional validation dataset. For example, Zhang et al. [51] assumes that the parameter server possesses some unlabeled datasets for voting, which compromises data privacy in federated learning and is impractical in a decentralized environment. In our work, we consider a decentralized FL scenario where benign clients can rely solely on their local clean data for backdoor defense. On the other hand, most methods focus on using knowledge distillation to repair a backdoor model [50,52], i.e., how to detoxify a poisoned model. In contrast, we consider how to use knowledge distillation to achieve the transfer of benign knowledge, preventing the introduction of backdoors during training rather than repairing a malicious model. This is evidently more efficient. Finally, in resource-constrained edge environments, clients often can only share parts of the model parameters, making method in [51], which relies on model inference on unlabeled dataset for consensus, infeasible. In our paper, we overcome the difficulty of model fragments being non-inferable by aggregating to obtain a complete twin model and use it as a teacher to facilitate the transfer of benign knowledge.

3. Model and problem definition

In this section, our FEEL with fragment-sharing system model, the general backdoor attack goal in FL, the problem definition for backdoor resilient FEEL with fragment-sharing and knowledge and capability of clients are introduced one by one.

3.1. FEEL system with fragment-sharing

In this paper, we consider a decentralized FEEL with fragment-sharing system comprising N clients, denoted by the set V. Unlike centralized FL with a parameter server for aggregation [53], the decentralized FL paradigm introduces heightened complexities for backdoor defense. Each client k possesses a local dataset D_k, local model θ_k and exchanges data over the network. However, owing to the user requirements or system constraints, each client k is restricted to transmitting a maximum of $\mathcal{B}_k$ bits of data during each communication process, which is different from traditional federated learning. This limitation is particularly reasonable, especially in the distributed training of large language models [54]. The limited edge network resources make it difficult for clients to share all model parameters, which promotes research based on sharing of model fragments. By training their own models locally and sharing the updates with others, all clients will achieve the following goal of our decentralized federated learning step by step:

$$\min_{\{\theta_1,\dots,\theta_N\}\in\mathbb{R}^d} \sum_{k=1}^{N} \frac{|D_k|}{|D|} f_k(\theta_k; D_k). \tag{1}$$

In the above equation, f_k represents the local loss function of each client k based on its local dataset D_k, such as cross entropy or mean square error loss. The overall dataset is denoted as $D = \{D_1 \cup D_2 \cup \cdots \cup D_N\}$ and θ_k represents a d-dimensional local model of client k.

To effectively optimize the objective (1) under the resource constrains, Wang et al. [7] adopt masking technology to achieve federated learning with partial model parameter sharing. Similar ideas are also reflected in the research of Qiao et al. [55]. Therefore, in this paper, we follow this idea and use masks to achieve decentralized FEEL with model fragment sharing. Specifically, each client is assigned with a initial local model θ_k^0. In each global round $t = 1, 2, \dots$, the client k first trains the local model on the local dataset, then transmitting a fragment of the model, conforming to the partial parameter constraints, to other clients. Finally, client k updates its local model upon receiving fragments from other clients. The procedural details are outlined as follows:

- **Local Training:** Each client k trains its own local model $\theta_k^{t,i}$ to optimize the local loss $f_k(\theta_k^i; D_k)$ on its dataset D_k for E epochs, i.e. $\theta_k^{t,i+1} = \theta_k^{t,i} - \eta_k^{t,i} \nabla f_k(\theta_k^{t,i}; D_k)$ for $i = 0, 1, 2, \dots, E-1$. $\theta_k^{t,i}$ denotes the ith local models of client k in global round t. Besides, $\eta_k^{t,i}$ is the corresponding learning rate in each iteration and $\nabla f_k(\theta_k^{t,i}; D_k)$ is the corresponding gradient of $f_k(\theta_k^{t,i}; D_k)$.
- **Fragment Sharing:** Each client k gets a model fragment $\hat{\theta}_k^t$ through mask m_k, i.e $\hat{\theta}_k^t = \theta_k^{t,E} \odot m_k$ and share $\hat{\theta}_k^t$ to other clients. Mask m_k is a binary matrix of the same size as θ_k^t with $\|m_k\|_0 \le \mathcal{B}_k$ and $\odot$ denotes the Hadamard product.
- **Fragment Aggregation:** Each client k aggregates the received model fragments and its local model to generate a new one that will be used in the next round local training. In a mathematical formulation, we have $\theta_k^{t+1,0} = \mathbf{Aggre}(\cup_{k=1}^{N} \{\hat{\theta}_k^t\})$. The $\mathbf{Aggre}(\,)$ is an abstract aggregation function here. Specifically, the parameter of each dimension in local model is the average of the corresponding dimension parameters of all its own and received fragments. Formally, the parameter of client k at eth dimension can be expressed as: $\theta_k^{t+1}(e) = (\theta_k^i(e) + \sum_{\substack{i\in V\setminus\{v\},\\ \mathcal{I}(\hat{\theta}_i^t)==1}} \hat{\theta}_i^t(e))/(c)$, where $c = \sum_{i\in V\setminus\{v\}} \mathcal{I}(\hat{\theta}_i^t)+1$ and $\mathcal{I}()$ is a judging function.

By repeating the training rounds for sufficient times, all clients obtain the high-accuracy model on its own dataset.

3.2. General backdoor attack in FEEL

We categorize the set of all clients V into two groups: the set of malicious nodes S_m and the set of benign clients S_b. Malicious clients possess a poisoned dataset, wherein specific triggers are added to some samples, and the real labels are modified. Generally, training on such a poisoned dataset results in a model with a backdoor. On the other hand, benign clients have a clean dataset, where all samples and labels remain unmodified. For each malicious client $k \in S_m$, during the fragment-sharing stage, it transmits a tampered model fragment $\tilde{\theta}_k^t$. Specifically, the non-zero positions in $\tilde{\theta}_k^t$ should align with those in the binary matrix m_k to meet system constraints. This intentional manipulation leads benign clients to obtain a backdoor model upon aggregating all the received model fragments.

In general, the backdoor model of a benign client behaves normally without triggers but exhibits malicious behavior when triggers are present. Let x and $\varphi(x)$ denote the clean and manipulated data sample, respectively. Similarly, y and $\tau(y)$ represent the corresponding true label and the target label that the malicious node aims to induce. The optimization goal of a general backdoor attack in FEEL with fragment-sharing is defined as follows:

$$\begin{aligned}
\min\ & \frac{1}{|S_b|} \sum_{\substack{k\in S_b,\\ \{x,y\}\in D_k}} f_k(\theta_k; \{x, y\}) + f_k(\theta_k; \{\varphi(x), \tau(y)\}) \\
\text{s.t. } & \theta_k^{t,i+1} = \theta_k^{t,i} - \eta_k^{t,i} \nabla f_k(\theta_k^{t,i}; D_k), i = 0, 1, 2, \dots, E-1, \\
& \hat{\theta}_k^t = \theta_k^{t,E} \odot m_k, \\
& \theta_k^{t+1,0} = \mathbf{Aggre}\{\cup_{k\in S_b}\{\hat{\theta}_k^t\}\} \cup \{\cup_{h\in S_m}\{\tilde{\theta}_h^t\}\}, \forall k \in S_b \\
& \|\hat{\theta}_b^t\|_0 \le \mathcal{B}_k, \hat{\theta}_b^t = \hat{\theta}_b^t \odot m_k, \forall b \in S_b, \\
& \|\tilde{\theta}_h^i\|_0 \le \mathcal{B}_h, \tilde{\theta}_h^i = \tilde{\theta}_h^i \odot m_h, \forall h \in S_m.
\end{aligned} \tag{2}$$

The main optimization objective of the attackers encompasses two main components. The first loss function encourages the victim to achieve high prediction accuracy on clean inputs, i.e. a backdoor model should have the same output with normal model on a clean sample. The second loss function aims for a high attack success rate when facing inputs with triggers, meaning the model produces the output desired by the attackers. Additionally, the first three constraints mirror the training process of benign clients in FEEL with fragment-sharing, describing the training process for benign clients. The latter two constraints reflect the constrictions of the system, that is, for both benign clients and malicious clients, they send a model fragment that satisfies this constraints. The specific fragments are determined by their respective masks.

In summary, in FEEL with fragment-sharing, the objective of the malicious client is to transmit a specific model fragment that satisfies the system constraints to the benign client during the fragment-sharing phase, so that when the model fragment is embedded into the local model of the benign client, can demonstrate backdoor features.

3.3. Problem definition

In contrast to backdoor attackers, our objective is to ensure that the local model trained by the benign client exhibits high prediction accuracy solely on clean inputs, without producing the result expected by the attacker for inputs with triggers. However, due to privacy restrictions in FL, clients can only share model parameters or gradients, but are not allowed to share relevant information such as local data or device capabilities, which makes benign clients hard to purify or filter received fragments from other clients. Given these constraints, benign clients must employ an efficient local training strategy to acquire knowledge from other clients while avoiding the introduction of potential backdoor into their local models. Consequently, the goal of backdoor resilient FEEL with fragment-sharing can be formulated as the following multi-objective optimization problem.

$$\begin{aligned}
& \min_{\theta_k\in\mathbb{R}^d} \{g_1(\theta_k), g_2(\theta_k)\}, \forall k \in S_b \\
\text{s.t. } & g_1(\theta_k) = -\mathbb{E}_{\{x,y\}\in D_k}[f_k(\theta_k; \{\varphi(x), \tau(y)\})], \\
& g_2(\theta_k) = \mathbb{E}_{\{x,y\}\in D_k}[f_k(\theta_k; \{x, y\})], \\
& \theta_k^{t+1,0} = \mathbf{Aggre}\{\cup_{k\in S_b}\{\hat{\theta}_k^t\}\} \cup \{\cup_{h\in S_m}\{\tilde{\theta}_h^t\}\}, \\
& \|\hat{\theta}_b^t\|_0 \le \mathcal{B}_b, \hat{\theta}_b^t = \hat{\theta}_b^t \odot m_k, \forall b \in S_b, \\
& \|\tilde{\theta}_h^i\|_0 \le \mathcal{B}_h, \tilde{\theta}_h^i = \tilde{\theta}_h^i \odot m_h, \forall h \in S_m.
\end{aligned} \tag{3}$$

Table 1
Important symbols.

Parameter	Definition
N	Number of clients
V	Set of clients
D_k	Dataset of client k
θ_k^t	Model parameters of client k
$\mathcal{B}_k$	Model constraint of client k
f_k	Local loss function of client k
η_k	Learning rate of client k
$\hat{\theta}_k, \tilde{\theta}_k$	Model fragment of client k
m_k	Mask matrix of client k
S_b	Set of benign clients
S_m	Set of malicious clients
E	Epoch number
B	A randomly sampled mini-batch
d	Dimensions of model parameters
γ	Coefficient of knowledge distillation loss
D_α	Parameter of Dirichlet function
$\{x, y\}, \{\varphi(x), \tau(y)\}$	Clean and poisoned sample
$\nabla f(\cdot)$	Gradient of function $f(\cdot)$
$\odot$	Hadamard product
$\| \|_0$	Zero norm
Aggre()	Aggregation function

The above two objectives reflect the goal of federated learning and the goal of resisting backdoor attacks, respectively. In this paper, our approach begins by aggregating a backdoor model and employing it as a teacher network for knowledge distillation on the local model. Leveraging the fact that the backdoor model behaves normally when faced with clean input, this strategy facilitates the transfer of global knowledge locally without introducing a backdoor.

3.4. Knowledge and capability of clients

For a malicious client, it can obtain a poisoned data set through data poisoning. Therefore, it can train on the poisoned dataset or acquire a backdoor model. During the fragment-sharing stage, malicious clients can arbitrarily modify the parameters of the shared model fragment. For example, this manipulation may include using model replacement [56], ensuring that the model aggregated by benign clients ultimately contains a backdoor. Importantly, we grant malicious clients knowledge of the network topology as well as information about other clients, including other malicious clients, allowing for cooperative actions among clients. The model fragments shared by malicious clients must conform to the system restrictions, although there are no constraints on mask choosing. It is crucial to note that malicious clients are explicitly prohibited from controlling with the normal training process of other benign clients.

On the other hand, benign clients are assigned with only a clean local dataset and are entirely unknown about the network topology and the any information of other clients. Besides, they do not allow any information sharing between benign clients except model fragments. Similarly, the model fragments of benign clients must adhere to system constraints and satisfy $\cup_{k\in S_b}\{m_k\} = \mathcal{J}$, where $\mathcal{J}$ is a matrix of ones, ensuring that all local model parameters of benign clients can learn knowledge from other benign clients. Finally, all important symbols used in the paper are summarized in Table 1.

4. BE-FEEL with fragment-sharing

In this section, we introduce our backdoor resilient FEEL with fragment-sharing (BR-FEEL) approach in detail. We begin by discussing the primary challenges encountered in approach design and subsequently propose corresponding solutions to address these challenges. Finally, we give a comprehensive description of the BR-FEEL approach.

4.1. Challenges and solutions

In the design of backdoor resilient FEEL with fragment-sharing approach, the primary challenges arise from the following three aspects: Firstly, we consider a general backdoor attack adversary without imposing any restrictions on it, except for prohibiting control over benign clients during normal training. Besides, we do not require benign clients to possess any prior knowledge or cooperation capabilities. Our threat model is more powerful and places stronger constraints on benign clients compared to previous work. Secondly, evaluating the security of model fragments becomes challenging. Notably, most previous backdoor detection and defense work is centered around full model parameters, with few work on detecting or defending against model fragments that may introduce a backdoor. Thus, the FEEL with fragment-sharing scenarios heighten challenges for our backdoor defense. Finally, how to migrate the knowledge from the fragment model to the local model without introducing a backdoor is a crucial consideration in the approach design.

Challenge 1: In this paper, we aim to defend against general backdoor attack in FEEL with fragment-sharing, as described in objective (2), where a malicious client strategically shares a designed model fragment to introduce a backdoor into the model aggregated by other benign clients. However, in our FEEL with fragment-sharing approach, benign clients lack prior knowledge of other clients and are not allowed to collaborate directly with each other. This constraint renders previous approaches, which are based on prior knowledge [10–12] or consensus among benign clients [57,58], ineffective. In summary, the unpredictability of malicious client behavior and the privacy requirements of FL pose heightened challenges for the design of our backdoor defense.

Challenge 2: In FEEL with fragment-sharing, clients only exchange a single fragment containing some model parameters. After receiving fragments from other clients, a benign client lacks the capability to verify the reliability of these fragments through direct conduct inference [9]. Additionally, malicious model fragments will only show abnormal behavior when they are embedded in the local model. Benign clients need to test the embedding of multiple combinations of fragments to prevent the conspiracy behavior of malicious clients, where only a specific combinations of multiple malicious fragments are embedded can get a backdoor model. However, validating the combination of fragments involves an exponentially computing cost, which is intolerable for the client. Consequently, identifying malicious fragments and mitigating their effects becomes extremely challenging.

Challenge 3: Effectively transferring knowledge from model fragments shared by other clients to the local model is a key consideration of benign clients. On the one hand, model fragments are only available after being embedded in the local model. On the other hand, embedding model fragments introduces the hidden danger of backdoor. In the FEEL with fragment-sharing proposed in [7], the benign client directly aggregates the model fragments and trains them locally to achieve knowledge transfer. But in the existence of backdoor attack, we need to find a safe way to migrate knowledge, which is very challenging.

Our Solutions: Since we are considering a general backdoor defense, we do not impose too many restrictions on malicious clients and do not give any prior knowledge to benign clients. Therefore, in our defense approach, our basis is the general goal of the backdoor attack, that is, predictive capability of the backdoor model on a clean dataset should be high, while simultaneously yielding desired outcomes for the attacker when confronted with input with a trigger. At the same time, as mentioned in the above challenges, it is almost impossible to verify model fragments, making it hard to achieve safe global knowledge transfer. Therefore, we first obtain a twin model using the normal model aggregation process. Due to the existence of malicious clients, the twin model is likely to have a backdoor. However, considering that the backdoor model produces normal results on the clean dataset of benign clients, we use the twin network as the teacher model to perform knowledge distillation on the local model. Specifically, the local model

learns from the true labels of the local dataset and simultaneously imitates the output of the teacher network to obtain global knowledge. Through knowledge distillation, we effectively transfer global knowledge to the local model without introducing a backdoor, strengthening the resilience of our defense approach.

4.2. Algorithm description

For each benign client $k \in S_b$, its process in BR-FEEL is given in Algorithm 1. Similar to the Vanilla FEEL with fragment-sharing, BR-FEEL approach can also be divided into three main stages. Initially, the benign client k utilizes the twin network θ_t as the teacher model to perform knowledge distillation on the local model, involving E local epochs. Each epoch employs multiple mini-batch stochastic gradient descents on the complete local dataset D_k. Specifically, for a non-repeating mini-batch B randomly sampled from D_k, the local objective is expressed as follows:

$$\begin{aligned} &\min \mathcal{F}_k(\theta_k) = (1-\gamma)\cdot f(\theta_k; B) + \gamma f_{KD}(\theta_k; B) \\ &\text{s.t. } f(\theta_k; B) = \frac{1}{|B|}\sum_{\{x,y\}\in B} \mathcal{L}_C(\theta_k(x), y), \\ &\quad f_{KD}(\theta_k; B) = \frac{1}{|B|}\sum_{\{x,y\}\in B} \mathcal{L}_R(z(\theta_k; x), z(\theta_t; x); T). \end{aligned} \tag{4}$$

In the above equation, γ is a hyperparameter adjusting the weight between local learning and knowledge distillation. Here, $z(\theta; x)$ represents the last-layer output vector of deep neural network θ when processing the input x, known as soft logits. T is the temperature coefficient, softening the logits extract more informative dark knowledge from the teacher model [36]. $\mathcal{L}_c$ is the classification loss function, such as cross-entropy. $\mathcal{L}_R$ reflects the discrepancy between the two results, often measured using Kullback–Leibler (KL) divergence. In the local knowledge distillation stage, the benign client continuously optimizes the local model θ_k to achieve the goal in (4). Local model needs to simultaneously learn local knowledge, enhancing performance on the local dataset, and imitate the teacher network's output results to transfer global knowledge.

Next, in the model fragment-sharing stage, the benign client k acquires a model fragment based on the latest local model and its mask, sharing it with other clients. Notably, this stage is the same with that in the FEEL with fragment-sharing.

Finally, in the model fragment aggregation stage, the benign client aggregates received model fragments into a new twin network. For each dimension e of the local model parameters, the client averages the parameters of the eth dimension with those included in the received sparse model. The algorithm employs the function $\mathcal{I}()$ to determine the number of participants sharing the eth dimension parameter, where $\mathcal{I}()$ is a function judging whether the input is 1. The average parameters for the eth dimension are then computed.

Compared with vanilla FEEL with fragment-sharing, BR-FEEL primarily differs in the local training process and model aggregation stage. In the local training process, BR-FEEL does not directly use the aggregated model for training on the local dataset. Instead, it adopts a twin network as the teacher network for knowledge distillation, allowing the local model to learn both local and global knowledge simultaneously. In the model aggregation stage, BR-FEEL does not employ the aggregated model as the starting point for the next local training round. The aggregated model serves as a new twin network for the subsequent round of local training. Through FL iterations, the global knowledge of the twin network steadily improves, helping to enhance the global knowledge of local model. Although the twin network is a backdoor model, since the benign client does not have poisoning data sets, the twin network can always give a good output result for an input without a trigger, which implies benign global knowledge can be acquired by local model by imitating the twin model.

Algorithm 1: BR-FEEL for Benign Client k

Input: clean dataset D_k, mask m_k, initial model $\theta_k^{0,0}$, initial twin model θ^0
Output: a clean local model without backdoor θ_k

for $t = 0, 1, 2, \ldots$ **do**
  for $i = 0, 1, 2, , \ldots E-1$ **do**
    for *each mini-batch* $B \in D_k$ **do**
      $\theta_k^{t,i} = \theta_k^{t,i} - \eta_k^{t,i} \nabla \mathcal{F}_k(\theta_k^{t,i}; B)$, where $\mathcal{F}_k$ is defined in (4);
    $\theta_k^{t,i+1} = \theta_k^{t,i}$;
  // Local knowledge distillation stage
  $\hat{\theta}_k^t = \theta_k^{t,E} \odot m_k$;
  Share $\hat{\theta}_k^t$ to all other clients;
  Receive $\hat{\theta}_i^t$ from other client i for $i \in V \setminus \{k\}$;
  // Model fragment sharing stage
  for *each dimension* $e = 0, 1, 2, \ldots, d-1$ **do**
    count $= \sum_{i \in V \setminus \{v\}} \mathcal{I}(\hat{\theta}_i^t) + 1$;
    $\theta_k^{t+1}(e) = (\theta_k^t(e) + \sum_{i \in V\setminus\{v\}, \mathcal{I}(\hat{\theta}_i^t)==1} \hat{\theta}_i^t(e))/(\text{count})$;
  // Model fragment aggregation stage

Table 2
A summary of important settings in experiment.

Dataset	CIFAR-10 [18], GTSRB [19]
Model	ResNet-34 [59], MobileNetV2 [60]
Attack Baselines	BadNet [20], Blend [21], Dynamic [22] Trojan [23], Adaptive_patch [24]
Defense Baselines	No Defense [7], Median [13], Geometric Median [25] Norm Clipping [26], BR-FEEL
FEEL Settings	$N = 10, E = 3, \epsilon \in \{0.1, 0.2, 0.5, 0.9\}$ $D_\alpha \in \{0.1, 1, 10, 1000\}, \eta = 0.0001$

5. Experiment

In this section, the numerical results of our BR-FEEL approach are presented. We comprehensively evaluate the performance of BR-FEEL under different attack methods and visualize the key results in the experiment. At the same time, the impact of proportion of malicious clients and the data distribution are also discussed in this section.

5.1. Experiment settings

Our entire experiment implemented using a Python program with the support of PyTorch [61], aiming to perform a computer vision classification task through federated learning. All experiments are conducted on a Linux machine equipped with 6 NVIDIA GeForce RTX 4090s and 192 GB main memory. The implementation is carried out in Python 3.9, utilizing CUDA for parallel computing [62]. The important experimental parameter settings are given as follows and summarized in Table 2.

Dataset and Model. Two common datasets, CIFAR-10 [18] and GTSRB [19], are considered. The training of the image classification adopts the ResNet-34 [59], a widely used deep residual network. It is defined by calling package *torchvision.models.resnet34*. To compare the impact of different models on the experimental results, we also considered the MobileNet-V2 network [60], which is a lightweight network commonly deployed on mobile and embedded devices.

Attack Baselines. To facilitate malicious clients in executing backdoor attacks on benign clients, they obtain a backdoor model through data poisoning. Five effective data poisoning methods, namely BadNet [20], Blend [21], Dynamic [22], Trojan [23], and Adaptive_patch

(a) Original. (b) BadNet. (c) Blend. (d) Dynamic. (e) Trojan. (f) Adaptive_patch.

Fig. 3. Visualization result of different data poison methods.

Table 3
Model fragment distribution in FL when $\epsilon = 0.2$. The bold part represents the information of malicious clients. During the training process, benign clients update all parameters, while malicious clients fine-tune the specified parameters.

Parameters partitioning	Training parameters	Client ID
Conv1, Layer1, Layer3	All Layers	2, 6
Bn1, Layer2, Layer3	All Layers	3, 7
Conv1, Layer1, Layer4	All Layers	4, 8
Layer2, Layer4, FC	All Layers	5, 9
Layer1, Layer3, FC	**Conv1, Layer1, Layer3, Layer4, FC**	**0**
Conv1, Layer3, Layer4	**Conv1, Layer1, Layer3, Layer4, FC**	**1**

[24], are considered in this paper. A visualization result of different data poison method on a sample in CIFAR-10 is shown in Fig. 3. Specifically, these methods embed specific triggers in samples from the clean dataset and tamper with their true labels, which can embed specific triggers for samples in the clean dataset and tamper with their true labels. The data poisoning rate is set to 20%, and labels are modified following the all-to-one pattern, wherein the real labels of all poisoned samples were altered to the same target label. Considering that in FEEL with fragment-sharing, malicious clients can only share some parameters of the model, using only data poisoning may reduce the effectiveness of backdoor attacks. In order to improve the effectiveness of the attack, malicious clients are allowed to control the local training process. Specifically, after receiving model fragments from other clients, a malicious client k will fine-tune only the parameters with their corresponding values in $\cup_{i \in S_m} m_i$ equal to 1, while freezing the parameters of other parts [63]. This approach can effectively embed the backdoor in multiple model fragments of the malicious client.

Defense Baselines. Considering that almost no existing work considers how to perform backdoor defense under FEEL with fragment-sharing, we carefully selected some In-AD defense strategies that can be directly transferred to this scenario. Because the fragment-sharing and decentralization fails most Pre-AD and Post-AD defense strategies. Firstly, Vanilla FEEL [7] is considered for a comprehensive comparison of effects before and after defense. Additionally, Median [13], Geometric Median [25], and Norm Clipping [26] are introduced for fragment backdoor defense. Both Median and Geometric Median employ robust aggregation strategies. Specifically, when a benign client receives fragments shared from another client, median and geometric median statistics are used for each dimension of the model to obtain a robust result by removing extreme values. Norm Clipping, on the other hand, is a defense strategy based on differential privacy. When a benign client shares its model fragment, it first applies norm regularization and then adds some Gaussian noise to perturb the shared parameters. Finally, BR-FEEL defense, based on knowledge distillation, is implemented to comprehensively compare the performance of these defense strategies under different attacks.

FEEL Settings. A decentralized FEEL scenario is considered, comprising 10 clients, with a fully connected network utilized between clients to share model fragments. The proportion of malicious clients ϵ is varied at values of $0.1, 0.2, 0.5, 0.9$. Each client use the Dirichlet function with parameter $D_\alpha = 1, 5, 10, 1000$, to obtain a subset of the entire dataset. The parameter D_α reflect the heterogeneity of the client data distribution, with smaller values indicating stronger data distribution heterogeneity. All clients undergo 20 rounds of global training, with each client performing 3 local iterations and a learning rate of 0.0001. Finally, the model fragment distribution for each client is summarized in Table 3.

Performance Metrics. The attack success rate (ASR) and clean data accuracy (CA) [9] are chosen as metrics to evaluate the performance of backdoor defense. ASR represents the probability that an input with a trigger is successfully predicted as the target class specified by the attacker. CA, on the other hand, denotes the probability that clean input samples without triggers are correctly predicted as their true classes.

5.2. Overall performance of BR-FEEL

In this section, we test the performance of BR-FEEL in detail under different attack methods, different amount of shared parameter-fragment, different attacker proportions, and different data distribution situations.

Overall Numerical Results. In this part, we conduct a comprehensive evaluation of the performance results of the aforementioned defense baselines under different attack methods and datasets. Specifically, we set the proportion of malicious clients ϵ to 0.2, designating clients 0 and 1 as backdoor attackers. All clients use the Dirichlet function with $D_\alpha = 10$ to obtain a subset of the CIFAR-10 and GTSRB datasets. After 20 rounds of the global training process, the average final ASR and CA results of all benign clients are presented in Table 4.

From the results of CIFAR-10 on ResNet-34, Vanilla FL, Median and Geometric Median all achieve CA values exceeding 93% under different attack methods. However, these methods prove ineffective against BadNet, Dynamic, Trojan, and Adaptive_patch attacks, resulting in ASR values exceeding 97%. Such poor performance is unacceptable, indicating that the aforementioned three methods are not effective in resisting most backdoor attacks. Notably, in the Blend attack, Median reduces the ASR by approximately 3.5% compared to Vanilla FEEL, while Geometric Median increases the ASR by about 10% compared to Vanilla FEEL. This highlights that defense strategies based on robust aggregation may only be effective under specific attack methods and data distributions. Norm Clipping, on the other hand, exhibits a low ASR of no more than 15% under five attack methods, especially under the Dynamic attack, achieving an extremely low ASR of 1.225%. However, the introduction of Gaussian noise in differential privacy also compromises the usability of model, with a CA of no more than 46% under five attack methods. Compared to the first three defense strategies, ASR and CA of Norm Clipping are reduced by at least 85% and 48% on average, respectively. In contrast, our BR-FEEL achieves the lowest ASR of no more than 2.5% and a CA of more than 87.5% under five attacks. Compared with the other four methods, it achieves an average ASR of 1.85% and a clean test rate of 87.67%, achieving a balance between defense and usability.

Similar trends are observed in GTSRB. Although Vanilla FEEL, Median, and Geometric Median achieve a high average CA of more than 96%, their ASR also exceeds 97%, raising concerns about the security. Surprisingly, Norm Clipping, while also achieving a lower ASR of 11.711%, shows a significant drop in CA. Compared to the

Table 4
A overall performance of different defense baselines.

Dataset	Model	Attack	Vanilla FEEL		Median		Geometric median		Norm clipping		BR-FEEL	
			ASR ↓	CA ↑	ASR ↓	CA ↑	ASR ↓	CA ↑	ASR ↓	CA ↑	ASR ↓	CA ↑
CIFAR-10	MobileNet-V2	BadNet	64.567	71.708	59.022	**73.301**	58.046	72.504	8.908	13.529	**2.281**	65.298
		Blend	29.661	**72.656**	23.072	72.503	24.072	72.443	13.615	16.771	**2.424**	66.221
		Dynamic	5.796	**72.068**	17.799	71.866	9.361	71.048	2.949	14.048	**2.231**	67.224
		Trojan	67.276	**73.195**	58.481	72.268	64.551	72.611	20.704	17.169	**2.201**	66.547
		Adaptive_patch	29.751	71.981	46.061	**72.849**	47.694	71.063	20.844	15.824	**2.569**	66.485
		Average	39.4102	72.322	40.877	**72.558**	70.745	71.934	13.404	15.472	**2.341**	66.355
	ResNet-34	BadNet	99.999	93.466	99.935	93.383	100.000	**94.049**	11.579	41.246	**1.889**	87.758
		Blend	79.793	93.381	76.249	92.443	98.129	**94.136**	5.839	45.536	**2.435**	87.566
		Dynamic	99.451	93.532	99.161	93.459	99.919	**99.901**	1.225	46.391	**0.832**	87.581
		Trojan	99.944	93.213	99.949	92.511	100.000	**94.336**	9.439	45.136	**2.311**	87.885
		Adaptive_patch	97.318	93.331	97.351	92.822	100.000	**93.831**	14.704	43.912	**1.772**	87.558
		Average	95.301	93.385	94.529	92.924	99.611	**95.251**	8.558	44.444	**1.848**	87.670
GTSRB	MobileNet-V2	BadNet	89.641	88.112	98.474	86.269	82.786	**89.021**	9.695	18.647	**2.747**	81.645
		Blend	29.096	87.561	65.319	**89.129**	74.866	88.139	**1.546**	9.572	2.738	81.946
		Dynamic	35.405	88.521	47.184	**89.906**	50.469	88.367	4.201	8.441	**2.213**	81.952
		Trojan	79.983	86.961	77.551	87.027	93.185	**88.524**	15.197	5.866	**2.566**	80.124
		Adaptive_patch	76.351	88.276	76.281	89.329	91.105	**89.529**	12.454	6.749	**2.881**	80.993
		Average	62.024	87.886	72.962	88.332	78.482	**88.716**	8.583	9.819	**2.629**	81.332
	ResNet-34	BadNet	87.521	**98.708**	100.000	97.898	100.000	97.309	13.339	17.725	**1.244**	80.795
		Blend	96.703	**98.675**	98.004	97.491	98.687	96.731	18.713	15.521	**2.083**	82.966
		Dynamic	99.997	98.263	99.998	95.334	99.734	**98.371**	7.739	15.336	**2.148**	81.952
		Trojan	100.000	**98.793**	100.000	98.012	100.000	98.611	9.142	15.498	**1.452**	83.256
		Adaptive_patch	99.978	**98.234**	96.967	97.913	99.921	97.661	9.621	16.341	**0.607**	81.786
		Average	96.840	**98.535**	98.994	97.330	99.668	97.665	11.711	16.084	**1.507**	82.151

Table 5
Model fragment distribution of ResNet-34 under small parameter amounts of the shared fragments.

Parameters partitioning	Training parameters	Client ID
Conv1, Layer1	All Layers	2, 6
Bn1, Layer3	All Layers	3, 7
Layer2, Layer4	All Layers	4, 8
Layer4, FC	All Layers	5, 9
Layer3, FC	**Conv1, Layer3, Layer4, FC**	**0**
Conv1, Layer4	**Conv1, Layer3, Layer4, FC**	**1**

Table 6
Model fragment distribution of ResNet-34 under large parameter amounts of the shared fragments.

Parameters partitioning	Training parameters	Client ID
Conv1, Layer1, Layer3, FC	All Layers	2, 6
Bn1, Layer2, Layer3, FC	All Layers	3, 7
Conv1, Layer1, Layer4, FC	All Layers	4, 8
Layer2, Layer3, Layer4, FC	All Layers	5, 9
Layer1, Layer2, Layer3, FC	**Conv1, Bn1, Layer1, Layer2, Layer3, Layer4, FC**	**0**
Conv1, Bn1, Layer3, Layer4	**Conv1, Bn1, Layer1, Layer2, Layer3, Layer4, FC**	**1**

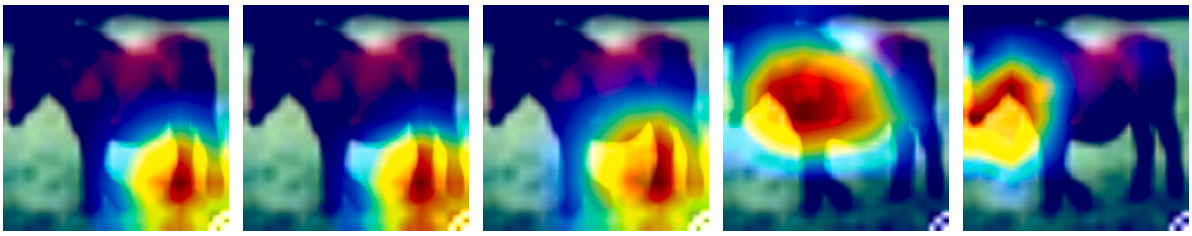
average CA in CIFAR-10, there is a decrease of approximately 28%. Nevertheless, our BR-FEEL consistently exhibits the lowest ASR with an average of 1.507%, while achieving a high CA with an average of 82.151%. These results demonstrate that BR-FEEL effectively resists various attack methods and displays robust adaptability across different datasets.

Similar results are also verified under MobileNetV2. BR-FEEL maintains an extremely low ASR of about 2% on the CIFAR-10 and GTSRB datasets. This shows that our BR-FEEL is a more general backdoor defense strategy that does not depend on a specific network model. However, due to the reduction in the number of network parameters, the CA of all defense baselines has dropped significantly.

Visualization results of BR-FEEL. To give a more intuitive comparison of the effectiveness of various backdoor defense strategies, we randomly select a local model from both a benign client and a malicious client after global training on the CIFAR-10 dataset and using the BadNet backdoor attack method, with the same settings in Table 4. We employ class activation mapping (CAM) [64] to visualize the feature attention distribution of these two models when facing a sample with a trigger. The results are depicted in 4 and 5.

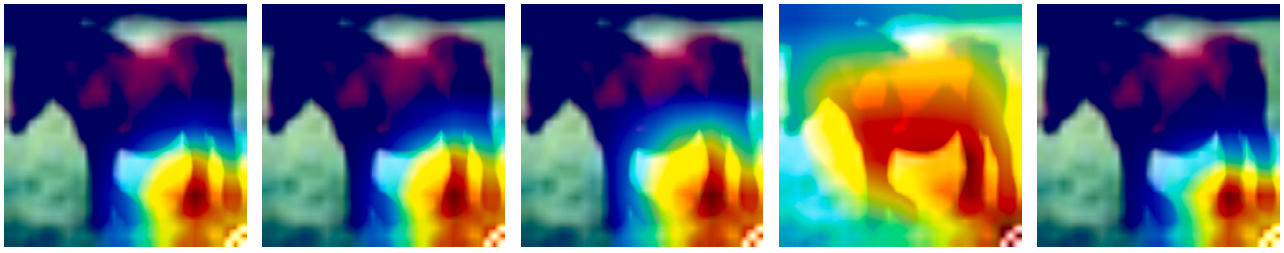
In CAM, regions with brighter color are considered more crucial features contributing to classification results. As observed in Fig. 4(a), 4(b), and 4(c), important features extracted from a benign client under Vanilla FEEL, Median, and Geometric Median defenses, when presented with an image of a horse containing a trigger, focus on the lower right corner of the image where the trigger is located. This suggests that a benign client under these defense strategies, when facing an input with a trigger, primarily classifies it based on trigger-related features, neglecting important benign features. Similar trends can be identified in the corresponding malicious counterpart images shown in Fig. 5(a), 5(b), and 5(c). This aberrant behavior is the primary reason why these three defense methods achieve high ASR under the BadNet attack. For Norm Clipping, as depicted in Fig. 4(d), the addition of Gaussian noise disrupts the attraction of benign clients towards trigger-related features in the lower right corner when faced with an input containing a backdoor. Simultaneously, Fig. 5(d) illustrates that the introduction of Gaussian noise also hampers the backdoor attack from the malicious client, resulting in extracted features that are not concentrated near the trigger when presented with an input containing a trigger. However, from a numerical perspective, the addition of Gaussian noise also diminishes the effectiveness of feature extraction by benign clients, leading to a low CA. In BR-FEEL, Fig. 4(e) indicates that for an image of a horse with a trigger, the features extracted by the benign client focus on the face, which are appropriate features for helping the model identify horses. Conversely, in Fig. 5(e), the extracted features of the malicious client are concentrated in the lower right corner of the image, indicating that the trigger is regraded more important than other parts of the horse, revealing a backdoor behavior.

Through visual comparison, it becomes evident that our BR-FEEL effectively prevents benign clients from learning incorrect features, thereby achieving an extremely low ASR.



(a) Vanilla FEEL. (b) Median. (c) Geometric Median. (d) Norm Clipping. (e) BR-FEEL.

Fig. 4. CAM Of benign client facing a sample with trigger under different defense methods.



(a) Vanilla FEEL. (b) Median. (c) Geometric Median. (d) Norm Clipping. (e) BR-FEEL.

Fig. 5. CAM Of backdoor client facing a sample with trigger under different defense methods.

Table 7
Overall performance of different backdoor defense baselines under varying amount of fragment sharing parameters on CIFAR-10.

Fragment setting	Vanilla FEEL		Median		Geometric median		Norm clipping		BR-FEEL	
	ASR(%)↓	CA(%)↑	ASR(%)↓	CA(%)↑	ASR(%)↓	CA(%)↑	ASR(%)↓	CA(%)↑	ASR(%)↓	CA(%)↑
Small	97.259	**92.825**	96.983	91.784	88.649	92.814	24.072	23.345	**1.757**	87.383
Middle	99.999	93.446	99.935	93.383	100.000	**94.049**	11.579	41.246	**1.889**	87.758
Large	100.000	**93.661**	99.992	93.452	100.000	93.586	26.357	20.823	**1.931**	88.023

Impact of different amount of shared parameter-fragment. In FL with fragment sharing, the amount of parameters in the shared fragments is a critical factor. Therefore, based on the fragment division of ResNet-34 in Table 3, we further reduced and increased the amount of parameters in the shared fragments. We refer to the parameter amounts of the shared fragments from small to large as Small, Middle, and Large, with the model divisions for Small and Large given in Tables 5 and 6. In this experiment, the Dirichlet parameter D_α is fixed at 10, the proportion of malicious clients ϵ is fixed at 0.2, and the attack method used is BadNet. Table 7 shows the ASR and CA results of various defense baselines under different sizes of shared fragment parameter amounts.

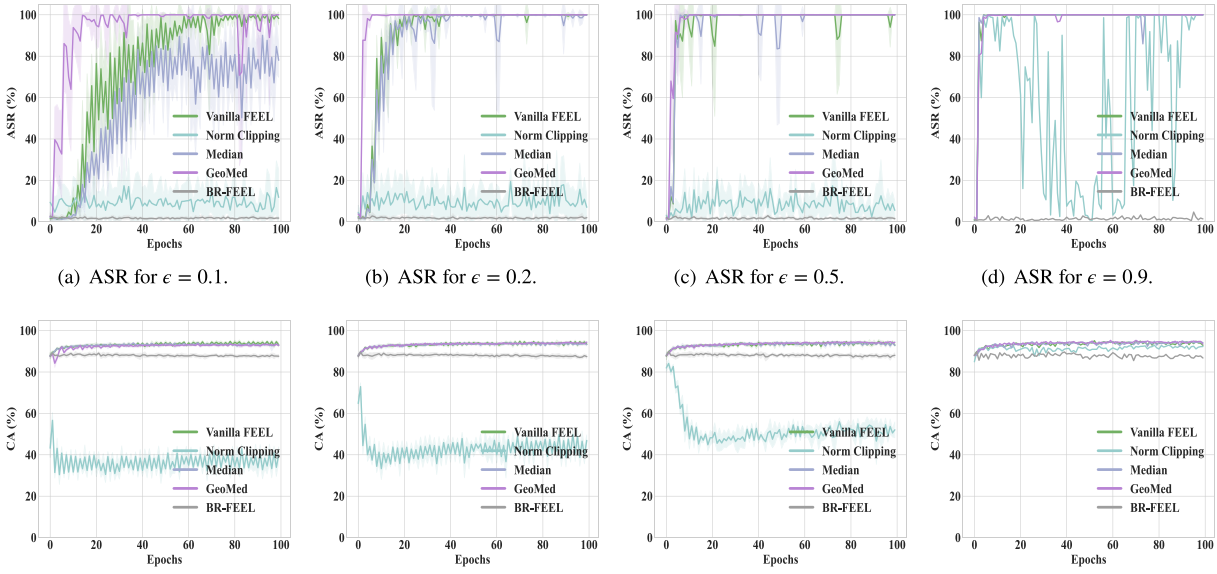
From the experimental results, it can be seen that as the amount of shared parameters increases, the ASR of benign clients also increases accordingly. This is because the attacker can share more parameters, meaning they can embed the backdoor in a larger parameter space. However, our BR-FEEL still demonstrates strong backdoor defense capabilities, with the ASR showing almost no change despite the increase in shared parameter amounts. Additionally, we note that as the shared parameter amount increases, the number of learnable parameters also increases, thus the CA of most defense baselines gradually increases as well.

Impact of various proportion of malicious clients. We explore the impact of the proportion of malicious clients, denoted as ϵ, on BR-FEEL. Specifically, we utilize the Dirichlet function with $D_\alpha = 10$ to partition the CIFAR-10 dataset, observing changes in ASR and CA as the number of training rounds increases for varying values of $\epsilon = 0.1, 0.2, 0.5, 0.9$. It is important to note that we designate clients with smaller IDs as attackers, as specified in Table 3.

As can be seen from Figs. 6(a), 6(b), 6(c) and 6(d), as the proportion of malicious clients ϵ increases, the convergence speed of ASR of Vanilla FEEL, Median and Geometric Median significantly accelerates. For example, when $\epsilon = 0.1$, Vanilla FEEL reaches nearly 100% ASR in 80 training rounds. However, with $\epsilon = 0.9$, this result is achieved in less than 10 training rounds. This trend indicates that as the proportion of malicious clients increases, the effectiveness of backdoor attacks is significantly enhanced, making backdoor defense more challenging. Similar results are observed for Norm Clipping. When the proportion of malicious clients does not exceed 0.5, Norm Clipping maintains a low ASR, averaging no more than 20%. However, with $\epsilon = 0.9$, its ASR converges to close to 100% after a large fluctuation. In contrast, our BR-FEEL maintains an ASR of no more than 2% under various proportions of malicious clients, demonstrating its robust backdoor resilience. As shown in Figs. 6(e), 6(f), 6(g), and 6(h), the CA of most defense strategies remains robust to varying proportions of malicious clients, achieving similar convergence speed and final performance. Specifically, Vanilla FEEL, Median, and Geometric Median can reach a CA close to 95% after about 10 rounds of training, which is approximately 8% higher than that of our BR-FEEL. Interestingly, as the proportion of malicious clients increases, the CA of Norm Clipping also rises accordingly.

In summary, the increase in the proportion of malicious clients enhances the effectiveness of backdoor attacks, with a relatively minor impact on CA. The experimental results also confirm that our BR-FEEL consistently achieves an extremely low ASR of no more than 2% under varying settings of malicious client proportions, effectively demonstrating its resilience against backdoor attacks.

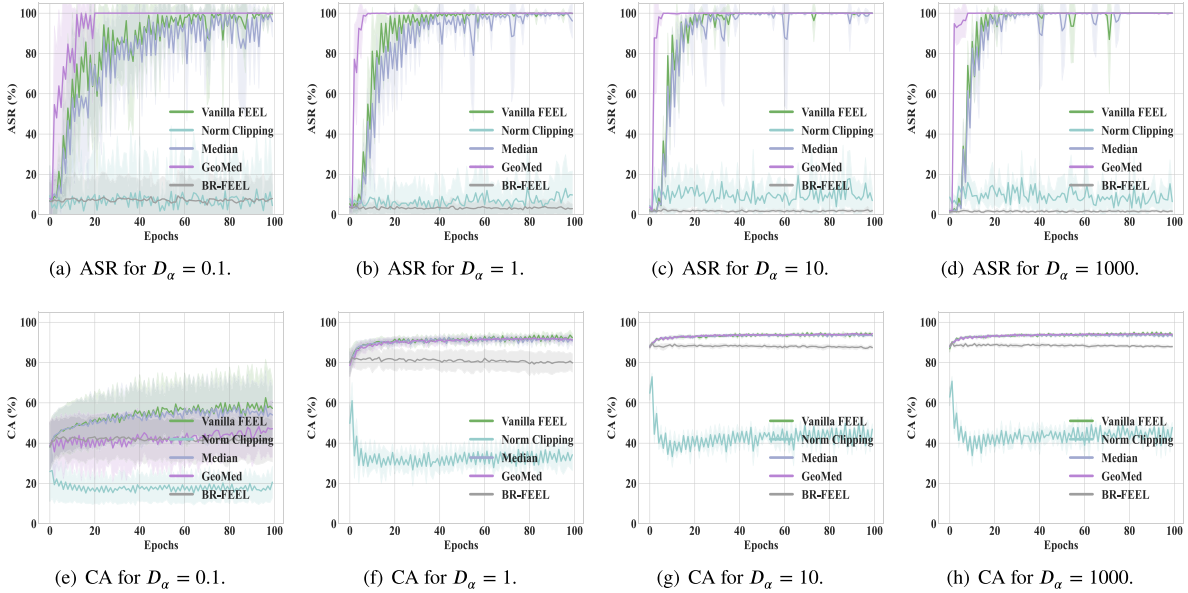
Impact of data distribution. In this part, we explore the impact of client data distribution on our BR-FEEL. Specifically, we fix the proportion of malicious clients, denoted as ϵ, to 0.2, meaning that clients 0 and 1 employ BadNet for backdoor attacks. By setting the Dirichlet parameter $D_\alpha = 0.1, 1, 10, 1000$, we can obtain distributions of CIFAR-10 data with different degrees of heterogeneity, and test the changes in ASR and CA of various defense strategies under these distributions with



(a) ASR for $\epsilon = 0.1$. (b) ASR for $\epsilon = 0.2$. (c) ASR for $\epsilon = 0.5$. (d) ASR for $\epsilon = 0.9$.

(e) CA for $\epsilon = 0.1$. (f) CA for $\epsilon = 0.2$. (g) CA for $\epsilon = 0.5$. (h) CA for $\epsilon = 0.9$.

Fig. 6. ASR and CA under various proportion of malicious clients ϵ.



(a) ASR for $D_\alpha = 0.1$. (b) ASR for $D_\alpha = 1$. (c) ASR for $D_\alpha = 10$. (d) ASR for $D_\alpha = 1000$.

(e) CA for $D_\alpha = 0.1$. (f) CA for $D_\alpha = 1$. (g) CA for $D_\alpha = 10$. (h) CA for $D_\alpha = 1000$.

Fig. 7. ASR And CA under various data distribution D_α.

the number of training rounds increasing. The experimental results are shown in Fig. 7.

As observed from Figs. 7(a), 7(b), 7(c) and 7(d), the enhanced heterogeneity of client local data distribution will lead to a decrease in the attack effectiveness of malicious clients. Specifically, under $D_\alpha = 0.1$, that is, under the strong heterogeneous data distribution, Median can achieve close to 100% ASR after about 90 rounds of training. However, the same result is achieved within no more than 20 rounds of training under $D_\alpha = 1000$, where the data is nearly independent and identically distributed. Similar situations are reflected in Vanilla FL and Geometric Median. For Norm Clipping and BR-FEEL, they show lower CA under different levels of data distribution. Specifically, Norm Clipping maintains an ASR of no more than 20% under four distributions. Our BR-FEEL can maintain an ASR of no more than 2% under four distributions, which reduces the ASR by 18% $\sim$ 98% compared to the other four defense baselines. This is because the attacker uses local data to conduct poisoning attacks. When the data distribution of the attacker and the defender is too different, the effect of data poisoning will be weakened. Therefore, when the data distribution of the client

becomes similar, the attacker is easier to conduct data poisoning, which enhances the backdoor attack capability.

Data distribution also has a great impact on CA. As can be seen from Figs. 7(e), 7(f), 7(g) and 7(h), as the data distribution becomes homogeneous, the convergence speed and final performance of CA are significantly improved, which aligns with the general behavior of federated learning in heterogeneous data environments [65]. For example, when $D_\alpha = 0.1$, the best-performing vanilla FEEL can only achieve 60% CA after 90 epochs of training. However, when $D_\alpha = 1000$, CA can converge to 95% after only 10 rounds of training.

On the other hand, we can see from Fig. 7(e) that the shadow area near the curve of each attack method is larger, which means that in the case of strong heterogeneity, the variance of CA between benign clients is large. On the other hand, in Fig. 7(e), the shadow area near the curve of each attack method is larger, indicating that in the case of strong heterogeneity, the variance of CA between benign clients is large. In Fig. 7(h), the shadow area is significantly reduced, showing that the heterogeneity of data distribution will lead to a larger variance in model CA between clients. Through a comparison between baselines, we find that under different data distributions, the three defense strategies Vanilla FEEL, Median, and Geometric Median can achieve the highest CA, slightly ahead of our BR-FEEL. As data homogeneity increases, this gap gradually narrows. At $D_\alpha = 1000$, the difference between the two is about 8%.

In summary, the strong heterogeneity of data distribution leads to the degradation of ASR and CA of other baselines. However, our BR-FEEL can still maintain the lowest ASR of $2\% \sim 7\%$ and high CA of $40\% \sim 90\%$ under various data distributions, fully demonstrating the effectiveness of its backdoor defense.

6. Conclusions

This paper addresses the vulnerability of backdoor attack in federated edge learning system with fragment-sharing. Specifically, we propose **b**ackdoor **r**esilient **fe**derated **e**dge **l**earning (BR-FEEL) approach, enabling benign clients to acquire global knowledge without introducing backdoor. In BR-FEEL, benign clients use twin models to integrate parameter fragments shared by other clients and as the teacher to conduct knowledge distillation on their clean dataset. Extensive experiments are conducted on CIFAR-10 and GTSRB using ResNet-34. Numerical results reveal that BR-FEEL significantly reduces the attack success rate by more than 90% compared to Vanilla FEEL. In future work, we will explore the problems of dynamic resources and privacy leakage faced by BR-FEEL in actual edge network deployment, and further explore the potential risks of adversarial attacks on BR-FEEL.

CRediT authorship contribution statement

Senmao Qi: Writing – review & editing, Writing – original draft, Methodology. **Hao Ma:** Validation, Methodology. **Yifei Zou:** Writing – review & editing, Supervision, Funding acquisition, Formal analysis. **Yuan Yuan:** Supervision. **Peng Li:** Supervision. **Dongxiao Yu:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62102232, 62122042, 62302247, Shandong Science Fund for Excellent Young Scholars (No. 2023HWYQ-007), and Postdoctoral Fellowship Program of CPSF under Grant GZC20231460.

References

[1] Zhuo Zhang, Yuanhang Yang, Yong Dai, Qifan Wang, Yue Yu, Lizhen Qu, Zenglin Xu, FedPETuning: When federated learning meets the parameter-efficient tuning methods of pre-trained language models, in: Annual Meeting of the Association of Computational Linguistics 2023, Association for Computational Linguistics (ACL), 2023, pp. 9963–9977.

[2] Zheng Lin, Guanqiao Qu, Qiyuan Chen, Xianhao Chen, Zhe Chen, Kaibin Huang, Pushing large language models to the 6g edge: Vision, challenges, and opportunities, 2023, arXiv preprint arXiv:2309.16739.

[3] Yangsibo Huang, Samyak Gupta, Zhao Song, Kai Li, Sanjeev Arora, Evaluating gradient inversion attacks and defenses in federated learning, Adv. Neural Inf. Process. Syst. 34 (2021) 7232–7241.

[4] Agrin Hilmkil, Sebastian Callh, Matteo Barbieri, Leon René Sütfeld, Edvin Listo Zec, Olof Mogren, Scaling federated learning for fine-tuning of large language models, in: International Conference on Applications of Natural Language To Information Systems, Springer, 2021, pp. 15–23.

[5] Jae Hun Ro, Theresa Breiner, Lara McConnaughey, Mingqing Chen, Ananda Theertha Suresh, Shankar Kumar, Rajiv Mathews, Scaling language model size in cross-device federated learning, 2022, arXiv preprint arXiv:2204.09715.

[6] Chaochao Chen, Xiaohua Feng, Jun Zhou, Jianwei Yin, Xiaolin Zheng, Federated large language model: A position paper, 2023, arXiv preprint arXiv:2307.08925.

[7] Yangyang Wang, Xiao Zhang, Mingyi Li, Tian Lan, Huashan Chen, Hui Xiong, Xiuzhen Cheng, Dongxiao Yu, Theoretical convergence guaranteed resource-adaptive federated learning with mixed heterogeneity, in: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2023, pp. 2444–2455.

[8] Haomin Zhuang, Mingxian Yu, Hao Wang, Yang Hua, Jian Li, Xu Yuan, Backdoor federated learning by poisoning backdoor-critical layers, 2023, arXiv:2308.04466.

[9] Thuy Dung Nguyen, Tuan Nguyen, Phi Le Nguyen, Hieu H Pham, Khoa D Doan, Kok-Seng Wong, Backdoor attacks and defenses in federated learning: Survey, challenges and future research directions, Eng. Appl. Artif. Intell. 127 (2024) 107166.

[10] Peva Blanchard, El Mahdi El Mhamdi, Rachid Guerraoui, Julien Stainer, Machine learning with adversaries: Byzantine tolerant gradient descent, Adv. Neural Inf. Process. Syst. 30 (2017).

[11] Luis Muñoz-González, Kenneth T. Co, Emil C. Lupu, Byzantine-robust federated machine learning through adaptive model averaging, 2019, arXiv preprint arXiv:1909.05125.

[12] Shiqi Shen, Shruti Tople, Prateek Saxena, Auror: Defending against poisoning attacks in collaborative deep learning systems, in: Proceedings of the 32nd Annual Conference on Computer Security Applications, 2016, pp. 508–519.

[13] Dong Yin, Yudong Chen, Ramchandran Kannan, Peter Bartlett, Byzantine-robust distributed learning: Towards optimal statistical rates, in: ICML, 2018, pp. 5650–5659.

[14] Mustafa Safa Ozdayi, Murat Kantarcioglu, Yulia R. Gel, Defending against backdoors in federated learning with robust learning rate, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, (10) 2021, pp. 9268–9276.

[15] Chen Wu, Xian Yang, Sencun Zhu, Prasenjit Mitra, Mitigating backdoor attacks in federated learning, 2020, arXiv preprint arXiv:2011.01767.

[16] C. Wu, S. Zhu, P. Mitra, Federated unlearning with knowledge distillation, 2022, arXiv preprint arXiv:2201.09441.

[17] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.

[18] Alex Krizhevsky, Geoffrey Hinton, et al., Learning multiple layers of features from tiny images, 2009.

[19] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition, Neural Netw. 32 (2012) 323–332.

[20] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, Siddharth Garg, Badnets: Evaluating backdooring attacks on deep neural networks, IEEE Access 7 (2019) 47230–47244.

[21] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, Dawn Song, Targeted backdoor attacks on deep learning systems using data poisoning, 2017, arXiv preprint arXiv:1712.05526.

[22] Tuan Anh Nguyen, Anh Tran, Input-aware dynamic backdoor attack, Adv. Neural Inf. Process. Syst. 33 (2020) 3454–3464.

[23] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, Xiangyu Zhang, Trojaning attack on neural networks, in: NDSS 2018, 2018.

[24] Xiangyu Qi, Tinghao Xie, Yiming Li, Saeed Mahloujifar, Prateek Mittal, Revisiting the assumption of latent separability for backdoor defenses, in: The Eleventh International Conference on Learning Representations, 2022.
[25] Krishna Pillutla, Sham M. Kakade, Zaid Harchaoui, Robust aggregation for federated learning, IEEE Trans. Signal Process. 70 (2022) 1142–1154.
[26] Ziteng Sun, Peter Kairouz, Ananda Theertha Suresh, H. Brendan McMahan, Can you really backdoor federated learning? 2019, arXiv preprint arXiv:1911.07963.
[27] Chaochao Chen, Xiaohua Feng, Jun Zhou, Jianwei Yin, Xiaolin Zheng, Federated large language model: A position paper, 2023, arXiv preprint arXiv:2307.08925.
[28] Frank Seide, Hao Fu, Jasha Droppo, Gang Li, Dong Yu, 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns, in: Fifteenth Annual Conference of the International Speech Communication Association, 2014.
[29] Sebastian U. Stich, Jean-Baptiste Cordonnier, Martin Jaggi, Sparsified SGD with memory, Adv. Neural Inf. Process. Syst. 31 (2018).
[30] Xiao Yang, Zhiyong Chen, Kuikui Li, Yaping Sun, Ning Liu, Weiliang Xie, Yong Zhao, Communication-constrained mobile edge computing systems for wireless virtual reality: Scheduling and tradeoff, IEEE Access 6 (2018) 16665–16677.
[31] Jed Mills, Jia Hu, Geyong Min, Multi-task federated learning for personalised deep neural networks in edge computing, IEEE Trans. Parallel Distrib. Syst. 33 (3) (2021) 630–641.
[32] Alysa Ziying Tan, Han Yu, Lizhen Cui, Qiang Yang, Towards personalized federated learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[33] Alysa Ziying Tan, Han Yu, Lizhen Cui, Qiang Yang, Towards personalized federated learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[34] Clement Fung, Chris J.M. Yoon, Ivan Beschastnikh, Mitigating sybils in federated learning poisoning, 2018, arXiv preprint arXiv:1808.04866.
[35] H Brendan McMahan, Daniel Ramage, Kunal Talwar, Li Zhang, Learning differentially private recurrent language models, 2017, arXiv preprint arXiv:1710.06963.
[36] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, Distilling the knowledge in a neural network, 2015, arXiv preprint arXiv:1503.02531.
[37] Sungsoo Ahn, Shell Xu Hu, Andreas Damianou, Neil D Lawrence, Zhenwen Dai, Variational information distillation for knowledge transfer, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9163–9171.
[38] Lu Yu, Vacit Oguz Yazici, Xialei Liu, Joost van de Weijer, Yongmei Cheng, Arnau Ramisa, Learning metrics from teachers: Compact networks for image embedding, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2907–2916.
[39] Jianping Gou, Baosheng Yu, Stephen J Maybank, Dacheng Tao, Knowledge distillation: A survey, Int. J. Comput. Vis. 129 (2021) 1789–1819.
[40] Jimmy Ba, Rich Caruana, Do deep nets really need to be deep? Adv. Neural Inf. Process. Syst. 27 (2014).
[41] Rafael Müller, Simon Kornblith, Geoffrey E. Hinton, When does label smoothing help? Adv. Neural Inf. Process. Syst. 32 (2019).
[42] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, Yoshua Bengio, Fitnets: Hints for thin deep nets, 2014, arXiv preprint arXiv:1412.6550.
[43] Byeongho Heo, Minsik Lee, Sangdoo Yun, Jin Young Choi, Knowledge transfer via distillation of activation boundaries formed by hidden neurons, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, (01) 2019, pp. 3779–3787.
[44] Defang Chen, Jian-Ping Mei, Yuan Zhang, Can Wang, Zhe Wang, Yan Feng, Chun Chen, Cross-layer distillation with semantic calibration, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, (8) 2021, pp. 7028–7036.
[45] Xingjian Li, Haoyi Xiong, Hanchao Wang, Yuxuan Rao, Liping Liu, Zeyu Chen, Jun Huan, Delta: Deep learning transfer using feature map with attention for convolutional networks, 2019, arXiv preprint arXiv:1901.09229.
[46] Junho Yim, Donggyu Joo, Jihoon Bae, Junmo Kim, A gift from knowledge distillation: Fast optimization, network minimization and transfer learning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4133–4141.
[47] Yige Li, Xixiang Lyu, Nodens Koren, Lingjuan Lyu, Bo Li, Xingjun Ma, Neural attention distillation: Erasing backdoor triggers from deep neural networks, 2021, arXiv preprint arXiv:2101.05930.
[48] Jun Xia, Ting Wang, Jiepin Ding, Xian Wei, Mingsong Chen, Eliminating backdoor triggers for deep neural networks using attention relation graph distillation, 2022, arXiv preprint arXiv:2204.09975.
[49] Kota Yoshida, Takeshi Fujino, Disabling backdoor and identifying poison data by using knowledge distillation in backdoor attacks on deep neural networks, in: Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security, 2020, pp. 117–127.
[50] Chengcheng Zhu, Jiale Zhang, Xiaobing Sun, Bing Chen, Weizhi Meng, ADFL: Defending backdoor attacks in federated learning via adversarial distillation, Comput. Secur. (2023) 103366.
[51] Jiale Zhang, Chengcheng Zhu, Chunpeng Ge, Chuan Ma, Yanchao Zhao, Xiaobing Sun, Bing Chen, BadCleaner: Defending backdoor attacks in federated learning via attention-based multi-teacher distillation, IEEE Trans. Dependable Secure Comput. (2024).
[52] Hanqi Sun, Wanquan Zhu, Ziyu Sun, Mingsheng Cao, Wenbin Liu, FMDL: Federated mutual distillation learning for defending backdoor attacks, Electronics 12 (23) (2023) 4838.
[53] Enrique Tomás Martínez Beltrán, Mario Quiles Pérez, Pedro Miguel Sánchez Sánchez, Sergio López Bernal, Gérôme Bovet, Manuel Gil Pérez, Gregorio Martínez Pérez, Alberto Huertas Celdrán, Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges, IEEE Commun. Surv. Tutor. (2023).
[54] Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, Robert McHardy, Challenges and applications of large language models, 2023, arXiv preprint arXiv:2307.10169.
[55] Jing Qiao, Shikun Shen, Shuzhen Chen, Xiao Zhang, Tian Lan, Xiuzhen Cheng, Dongxiao Yu, Communication resources limited decentralized learning with privacy guarantee through over-the-air computation, in: Proceedings of the Twenty-Fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, 2023, pp. 201–210.
[56] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, Vitaly Shmatikov, How to backdoor federated learning, in: International Conference on Artificial Intelligence and Statistics, 2020, pp. 2938–2948.
[57] Harsh Bimal Desai, Mustafa Safa Ozdayi, Murat Kantarcioglu, Blockfla: Accountable federated learning via hybrid blockchain architecture, in: Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy, 2021, pp. 101–112.
[58] Zonghang Li, Hongfang Yu, Tianyao Zhou, Long Luo, Mochan Fan, Zenglin Xu, Gang Sun, Byzantine resistant secure blockchained federated learning at the edge, IEEE Netw. 35 (4) (2021) 295–301.
[59] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[60] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.
[61] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al., Pytorch: An imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. 32 (2019).
[62] Jason Sanders, Edward Kandrot, CUDA by example: an introduction to general-purpose GPU programming, Addison-Wesley Professional, 2010.
[63] Ahmadreza Jeddi, Mohammad Javad Shafiee, Alexander Wong, A simple fine-tuning is all you need: Towards robust deep learning via adversarial fine-tuning, 2020, arXiv preprint arXiv:2012.13628.
[64] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra, Grad-cam: Visual explanations from deep networks via gradient-based localization, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 618–626.
[65] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, Zhihua Zhang, On the convergence of fedavg on non-iid data, 2019, arXiv preprint arXiv:1907.02189.

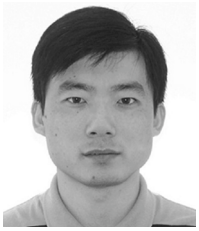
Senmao Qi received the B.E. degree from the School of Computer Science and Technology, Shandong University, Qingdao, China, in 2021. He is currently working toward the Ph.D. degree with the school of computer science and technology, Shandong University, Qingdao, China. His research interests include distributed machine learning, AI security and wireless network.

Hao Ma is currently a junior at the School of Computer Science and Technology, Shandong University, Qingdao, China. His interests include Federated Learning and AI Security.

Yifei Zou (Member, IEEE) received the B.E. degree in computer science and technology from Computer School, Wuhan University, Wuhan, China, in 2016, and the Ph.D. degree in computer science from the Department of Computer Science, The University of Hong Kong, Hong Kong, in 2020. He is currently an Assistant Professor with the School of Computer Science and Technology, Shandong University, Qingdao, China. His research interests include wireless networks, ad hoc networks, and distributed computing.

Yuan Yuan received the B.Sc. degrees from the School of Mathematical Sciences, Shanxi University in 2016, and the Ph.D. degree from the School of Computer Science and Technology, Shandong University, Qingdao, China, in 2021. She is currently a postdoctoral fellow at the Shandong University-Nanyang Technological University International Joint Research Institute on Artificial Intelligence, Shandong University. Her research interests include distributed computing and distributed machine learning.

Peng Li (Senior Member, IEEE) is a Senior Associate Professor in the University of Aizu, Japan. His research interests mainly focus on wired/wireless networking, cloud/edge computing, distributed AI systems, and blockchain. Dr. Li has authored or co-authored over 100 papers in major conferences and journals. Dr. Li won the 2020 Best Paper Award of IEEE Transactions on Computers. He serves as the chair of SIG on Green Computing and Data Processing in IEEE ComSoc Green Communications and Computing Technical Committee. Dr. Li is the guest editor of IEEE Journal of Selected Areas on Communications, the editor of IEEE Open Journal of the Computer Society, and IEICE Transactions on Communications. He is a senior member of IEEE.



Dongxiao Yu (Senior Member, IEEE) received the B.Sc. degree in mathematics from the School of Mathematics, Shandong University, Jinan, China, in 2006, and the Ph.D. degree in computer science from the Department of Computer Science, The University of Hong Kong, Hong Kong, in 2014. In 2016, he became an Associate Professor with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China. He is currently a Professor with the School of Computer Science and Technology, Shandong University. His research interests include wireless networks, distributed computing, and graph algorithms.