Neurocomputing 604 (2024) 128317

Contents lists available at ScienceDirect

Neurocomputing

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

Preventing harm to the rare in combating the malicious: A filtering-and-voting framework with adaptive aggregation in federated learning

Yanna Jiang [a,1], Baihe Ma [a,1], Xu Wang [a], Guangsheng Yu [b], Caijun Sun [c,*], Wei Ni [b], Ren Ping Liu [a]

[a] University of Technology Sydney, Sydney, 2007, NSW, Australia
[b] Data61, CSIRO, Sydney, 2122, NSW, Australia
[c] Zhejiang Lab, Hangzhou, 311121, Zhejiang, China

ARTICLE INFO

Communicated by G. Yu

Keywords:
Federated learning
Secure aggregation
Reinforcement learning

ABSTRACT

The distributed nature of Federated Learning (FL) introduces security vulnerabilities and issues related to the heterogeneous distribution of data. Traditional FL aggregation algorithms often mitigate security risks by excluding outliers, which compromises the diversity of shared information. In this paper, we introduce a novel filtering-and-voting framework that adeptly navigates the challenges posed by non-iid training data and malicious attacks on FL. The proposed framework integrates a filtering layer for defensive measures against the intrusion of malicious models and a voting layer to harness valuable contributions from diverse participants. Moreover, by employing Deep Reinforcement Learning (DRL) for dynamic aggregation weight adjustment, we ensure the optimized aggregation of participant data, enhancing the diversity of information used for aggregation and improving the performance of the global model. Experimental results demonstrate that the proposed framework presents superior accuracy over traditional and contemporary FL aggregation methods as diverse models are utilized. It also shows robust resistance against malicious poisoning attacks.

1. Introduction

Federated Learning (FL) [1] has garnered substantial attention as a potent solution for data protection and storage issues [2,3] that eliminates the need to share raw data and facilitates the global model to evolve by aggregating contributions from diverse participants. However, the distributed training model employed in FL raises concerns regarding security challenges [4] because of its vulnerabilities to potential attacks such as data poisoning and backdoor attacks. To mitigate security concerns, FL aggregation algorithms, such as Krum [5], median [6], and trimmed mean [6], implement Byzantine Fault Tolerance (BFT) [7] by selectively aggregating shared models. While effective in defending against attacks, these methods often discard unique and valuable data from diverse participants. This reduction in the diversity of aggregated information constrains the potential of the global model to fully leverage rich, varied datasets and adversely affects the robustness and generalization ability of the global model [8].

In real-world applications, the diversity of participant data, often non-independent identical (non-iid) distribution, sparse, and imbalanced [9,10], presents additional challenges in FL as the omission of models trained on distinctive datasets notably hampers the performance of the global model [11]. For instance, in tasks like fault diagnosis [12], the scarcity of fault data results in the issue of fragmented data islands, making the effective use of these rare datasets crucial for enhancing the generalization ability and accuracy of FL models. Thus, the inclusion of diverse data is beneficial and essential in FL secure aggregation to develop a robust and well-performed global model, which is overlooked in current research.

Moreover, the rare and valuable information inherent in small sample sizes may be diluted during the aggregation [13], making it challenging for the global model to learn their unique knowledge. Fixed aggregation weights based on the size of training datasets such as FedAvg [14] fail to meet the needs of non-iid scenarios. Consequently, the dynamic adjustment of aggregation weights becomes critical as it directly influences the utilization rate of diverse information, determining how effectively the FL system can incorporate and benefit from these distinct data contributions.

This paper is dedicated to enabling secure and diverse aggregation in an FL environment with malicious participants and non-iid training

[*] Corresponding author.
E-mail addresses: Yanna.Jiang@student.uts.edu.au (Y. Jiang), Baihe.Ma@uts.edu.au (B. Ma), Xu.Wang@uts.edu.au (X. Wang), Saber.Yu@data61.csiro.au (G. Yu), sun.cj@zhejianglab.com (C. Sun), Wei.Ni@data61.csiro.au (W. Ni), RenPing.Liu@uts.edu.au (R.P. Liu).
[1] Yanna Jiang and Baihe Ma contributed equally to this paper.

https://doi.org/10.1016/j.neucom.2024.128317
Received 25 April 2024; Received in revised form 10 July 2024; Accepted 3 August 2024
Available online 10 August 2024

data. Our work introduces a novel filtering and voting framework to refine the model evaluation in the aggregation process of FL. This framework effectively rejects malicious models while leveraging collective agreement to preserve the richness and variety of knowledge learned by the global model. A filtering layer is designed as a first line of defense, screening out potentially malicious or outlier contributions based on discrepancies in model behavior. A voting layer, operating independently and asynchronously from the filtering layer, is introduced to further assess the pre-screened models with a democratic voting process. The voting mechanism not only augments the resilience of the framework against malicious attacks such as data poisoning and backdoor attacks but also preserves valuable data contributions from diverse participants, especially from those with unique training data. Moreover, by integrating Deep Reinforcement Learning (DRL) [15], we dynamically adjust aggregation weights to prioritize the impact of models containing rare and valuable information on the global model and to coordinate the asynchronous outcomes of the proposed two layers. The DRL-based aggregation weight selection further enhances the diversity adaptability, and learning efficiency of the model aggregation.

The key contributions of our work are listed as follows.

(1) We propose a novel filtering-and-voting framework for secure and optimized performance in FL, consisting of a filtering layer based on model distance computation and a voting layer designed to assess models excluded by the filtering layer. The filtering layer defends against malicious attacks on model parameters and data poisoning, while the voting layer retains the contributions of honest participants with unique data to the global model.

(2) We evaluate the proposed framework analytically and provide a convergence bound to substantiate that the asynchronous setup between the filtering and voting layers does not compromise the convergence of the FL global model.

(3) We design a DRL-based dynamic aggregation weight selection for the optimized combination of outcomes from the voting and filtering layers. It effectively prioritizes learning special and important knowledge from participants with unique data and improves the generalization capabilities of the global model.

Experiments demonstrate that with MNIST dataset, the proposed framework can improve accuracy by 7.98% and 6.96%, compared to FedAvg [14] and multi-Krum [5], respectively; and improve accuracy by 6.78% compared to the state-of-the-art FL secure aggregation method like FLAME [16]. Moreover, under the label-flipping poisoning attacks [17], our framework can provide 100% resistance against malicious attacks and aggregate the valuable models provided by the participants with unique data with over 90% probability.

The rest of this paper is organized as follows. In Section 2, background and related works are reviewed. The proposed filtering-and-voting aggregation FL framework is presented in Section 3 with a mathematical analysis of the convergence. In Section 4, we discuss the DRL-based model for dynamically selecting aggregation weights. The practical feasibility of the proposed framework is evaluated through experimental in Section 5, followed by conclusions in Section 6.

2. Related work

FL is expanding into diverse real-world applications, intensifying the challenges associated with non-iid training data. Research indicates that global data imbalances in FL result in significant reductions in model accuracy [18]. In response, researchers are turning their attention to the sparse but crucial data resources present in imbalanced training data in FL.

To address the deterioration in accuracy caused by non-iid training data in FL, Astraea [18] was developed as a solution by incorporating mediators into the training process to achieve global data re-balancing. However, the integration of mediators within Astraea imposed a requirement on FL participants to disclose details about the distribution of their localized training data, potentially giving rise to newfound privacy apprehensions. In contrast, the research presented in [19] endeavored to determine the existence of data imbalance in FL through a monitoring mechanism, thereby avoiding the need for direct sharing of information regarding local data distributions. Within each iteration of FL, a monitor deduced the influence of individual classes on the global model and introduced a novel loss function, Ratio Loss, as a means to mitigate the challenges posed by both local and global data imbalances. The BalanceFL framework [20] divided the issues posed by imbalanced data into two discernible facets: local and global. To address the global issue of missing classes, it harnessed the mechanism of knowledge inheritance, while to tackle the local inter-class imbalance problem, it employed balanced sampling techniques. Through collaborative efforts from both parties, it demonstrated superior performance compared to prevailing FL models. These approaches paid attention to addressing the challenges associated with non-iid training data in FL, yet they did not account for the selection of participants or the adjustment of their corresponding aggregation weights.

To enhance the diversity of knowledge learned by the global model in non-iid training scenarios, further consideration has been given to the participant selection and the aggregation weight modification. Wang et al. [21] incorporated RL to analyze the training data distribution among FL participants based on the implicit connection between data distribution and trained model weights. This approach enabled an intelligent participant selection process in each FL round, effectively offsetting the bias caused by non-iid data and maintaining the diversity of aggregated information. Moreover, the ABAVG method was developed to improve the accuracy and convergence speed of global models in non-iid training data scenarios [13], where scarce information was magnified because of its greater improvements in accuracy. This method demonstrated that dynamically adjusted aggregation weights could significantly enhance server-side aggregation, offering substantial improvements over traditional methods with fixed aggregation weights. However, the methodologies fall short in addressing security challenges presented by Byzantine nodes, as the substantial variations between benign updates hinder the effective detection of malicious attacks and malicious intrusions such as backdoor attacks may not be reflected in accuracy. [22].

To achieve secure aggregation in FL, filtering and aggregation mechanisms have been proposed, assuming that the benign models are close to each other while malicious models are discernibly different from the benign ones, for instance, Krum [5] and median [6]. Peng et al. [23] incorporated a resampling strategy into the geometric-median-based aggregation algorithm. This addition served to diminish the influence of data variation on model performance, all the while enhancing security. DiverseFL [24] was designed to address the challenge of Byzantine behaviors with heterogeneous data distribution in FL. The center server in DiverseFL identified the Byzantine clients using the guiding gradient, computed with a small sample, to evaluate whether the update meets expectations. By comparing data against the samples owned by the target client rather than with other clients, DiverseFL advanced beyond the constraints of similarity-based approaches in handling heterogeneous and diverse data. This strategy provided fault resiliency of secure FL, nevertheless, it remains vulnerable to attacks such as data poisoning and backdoor attacks, because it relies on the assumption that clients are honest but faulty.

In [16], FLAME was introduced as a defense framework against malicious poisoning attacks and backdoor attacks. FLAME employed a clustering algorithm based on the cosine similarity to aggregate models that exhibit similar characteristics. FLAME also incorporated norm-based median clipping and proactive noise addition to enhance the resilience of the global model against variability in shared models caused by differing training data. These strategies enabled the FLAME framework to perform effectively in non-iid training data scenarios as it was less sensitive to such disparities. However, FLAME cannot fully detect attacks but only mitigate the effects of poisoned models.

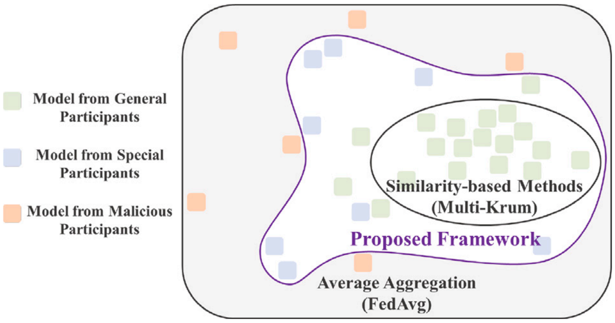

Model from General Participants
Model from Special Participants
Model from Malicious Participants
Similarity-based Methods (Multi-Krum)
Proposed Framework
Average Aggregation (FedAvg)

Fig. 1. Vector space of model parameters in FL with special participants and malicious participants. These special participants with unique data have distinctive training data information, diverging from that of general participants, while malicious participants attempt to compromise the global model through data poisoning or model poisoning attacks. The objective of the proposed framework is to facilitate secure aggregation in FL, protecting the global model of malicious participants, while optimizing the utilization of the shared models from special participants with unique data and minimizing valid information loss.

The processes of clipping and adding noise designed for robustness did compromise the ability of FLAME to capitalize on distinctive and crucial information, thereby limiting its effectiveness in preserving the diversity of valuable knowledge.

Overall, existing secure FL aggregation methods in non-iid training data scenarios achieved security by compromising on diversity. There remains a gap in secure aggregation for FL under non-iid training scenarios to simultaneously ensure accurate detection of malicious models and maximize the utilization of diverse information. In this paper, we introduce a novel filtering and voting framework for secure FL aggregation under non-iid training data sets. By incorporating a re-selection mechanism within the voting layer, the framework can include as many benign models as possible for aggregation and reject malicious models, thereby enriching the diversity of the aggregated models and enhancing the security and accuracy of the global model.

3. Filtering-and-voting aggregation framework

In this section, we introduce the proposed filtering-and-voting FL aggregation framework implementing the filtering-and-voting structure. We begin by outlining the scenarios and research motivations addressed in this paper, setting forth the objective of achieving secure aggregation in FL while preserving model diversity. We then detail the process of the proposed framework, which involves a selection by the filtering layer followed by a re-evaluation by the voting layer, coupled with dynamic aggregation weight selection based on DRL. This approach allows for the effective utilization of valuable shared models and provides resistance against malicious influences. Finally, we substantiate the convergence of the proposed framework through theoretical proof, demonstrating its robustness and efficacy.

3.1. System model

In this paper, we focus on an open FL scenario involving imbalanced training data and potential attackers. We classify FL participants into three groups: general participants, special participants with unique data, and malicious participants. Special participants have unique local training datasets, resulting in their shared models being distinct from those of general participants. Malicious participants seek to compromise the performance of the global model through data or model poisoning attacks. Here, we make the assumption that the number of malicious participants is fewer than half of the total participant count.

As shown in Fig. 1, trained models shared by general participants with similar training data demonstrate a high degree of similarity, resulting in a concentration within the model parameter vector space. In contrast, the shared models of special and malicious participants exhibit dissimilarity and remain distant from the models of general participants in the vector space. Average aggregation methods in FL, such as FedAvg [14], incorporate all of them directly for aggregation, lacking resistance against malicious attacks. Similarity-based aggregation methods in FL, such as Krum [5], selectively choose models clustered in the vector space for aggregation. This kind of method effectively excludes the models from malicious participants, but also discards those from special participants with unique data, losing valuable information for the advancement of global models.

There is clearly a need for an evolved FL aggregation method that can balance security and diversity by effectively assessing the contribution of shared local models to the FL global model and adjusting aggregation weights accordingly. As observed from Fig. 1, distinguishing between models from participants with unique data and those from malicious participants is challenging when relying solely on model distance or similarity. Consequently, the aim of this paper is to design a framework that dynamically evaluates and selects aggregation models. This framework seeks to learn from as many benign models as possible and fully utilize diverse training data to optimize the performance of the global model. At the same time, it provides protection against malicious models, effectively balancing security and performance enhancement.

3.2. Framework design and workflow

In addition to the filtering process in secure FL, the proposed framework introduces an additional layer, an accuracy-based re-selection mechanism called the voting layer. The filtering layer picks out distinctive models via distance calculations, and the voting layer assesses the value of models dropped in the filtering layer to improve the global model based on test accuracy. This strategy significantly enhances the selection process for aggregation models. Models from special participants with unique local training datasets can be selected back for aggregation to ensure the integration of precious knowledge and contribute to the global model, while the poisoned models supplied by malicious participants will be excluded from aggregation to reduce threats and risks. Moreover, the implementation of a DRL-based adaptive aggregation weight selection optimizes the performance of the aggregated global model. It effectively prioritizes the impact of models containing rare and valuable information and coordinates the outcomes of the filtering and voting layers.

In our previous work [25], we attempted to re-evaluate participants through the introduction of the "Think Tank", which significantly increased the time cost per FL epoch; i.e., it cost nearly triple the time required by the classical aggregation methods like Krum [5]. Moreover, it necessitated training for an extended number of epochs to observe any noticeable change in accuracy. Drawing from these experiences, this paper presents an enhanced approach by implementing asynchronous operations between the filtering and voting layers, significantly reducing the time required for each FL epoch and enhancing the feasibility of our proposed framework.

Assuming that there are N participants, denoted as $P_1, \cdots, P_i, \cdots, P_N$, each of which owns a local training dataset D_i and a testing dataset D_i^{test}, $i \in \mathbb{N}_{\mathbb{P}} = \{1, \ldots, N\}$, and one aggregator consists of committees that can be composed of participants or trusted servers. Due to the localized ownership of all datasets by individual participants, both model training based on the training set and accuracy determination based on the testing dataset are conducted in a distributed manner. Alongside the traditional participants and aggregator in classic FL, a new identity, that of "voters", is introduced. We consider the case where the voters consist of all participants here, and in future work, we plan to achieve dynamic voter selection by implementing a creditworthiness scoring mechanism. In the epoch r of FL, the local model shared by the participant P_i is denoted as M_i^r, and the global model is denoted as M_G^r. The processes of the proposed filtering-and-voting

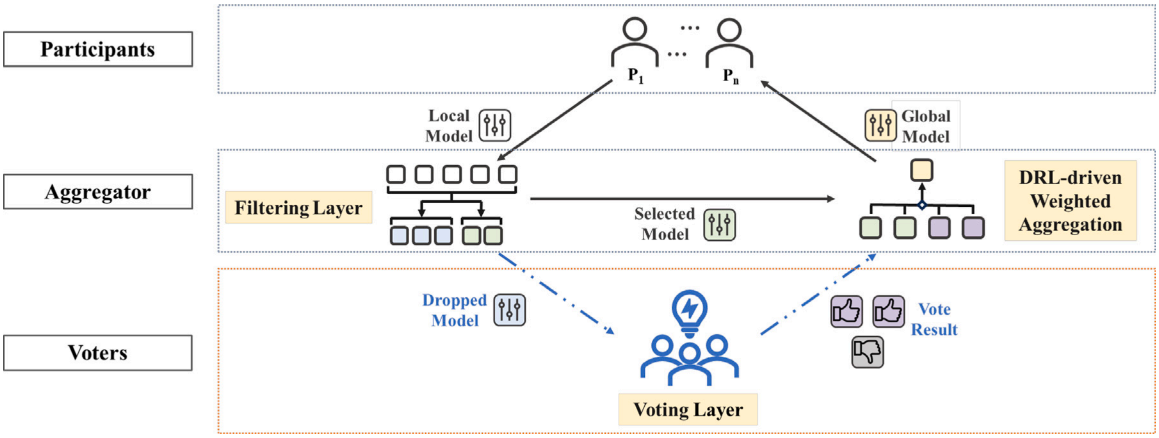


Fig. 2. Proposed filtering-and-voting aggregation framework with DRL-based weight selection in FL. It shows a five-step process to enhance performance and ensure security, including global model initialization, local model training, filtering layer for model selection, voting layer for model re-evaluation, and aggregation finalization. We assume that more than $1/2$ of the participants are benign and effective.

Algorithm 1 Filtering-and-Voting Aggregation Algorithm.

```
Require:
    Global Epoch $r$, Voting Layer Processing Epoch $r'$, Local Training Model $M_i^r$ from Participant $P_i$ ($i \in \mathbb{N}_{\mathbb{P}}$) with Training Dataset $D_i$ and Test Dataset $D_{Test_i}$, Filtering-Layer Coefficient $K$, List of Global Model in each epoch $\mathscr{L}_G$, List of Dropped Model from the Filtering Layer $\mathscr{L}_U$
Ensure:
    Global Model $M_G^r$.
    [Filtering Layer]
    $score_i^r = \sum_{x \in \mathbb{N}_{\mathbb{P}}} \|M_i^r - M_x^r\|^2$
    $M_B^r = \frac{\sum_{j=1}^K |D_{S_j}| M_{S_j}^r}{\sum_{j=1}^K |D_{S_j}|}$ ( $M_{S_j}^r$ has the K smallest $score_i^r$)
    Update $\mathscr{L}_U$ with models except $M_{S_j}^r$
    [Vote Layer]
    $P_i$ tests $M_{U_t}^{r'} \in \mathscr{L}_U$ and $M_G^{r'} \in \mathscr{L}_G$ on $D_{Test_i}$ for accuracy $Acc_{iU_t}^{r'}$ and $Acc_{iG}^{r'}$
    if $Acc_{iU_t}^{r'} \geq Acc_{iG}^{r'}$ then
        $I_{iU_t}^{r'} = 1$
    else
        $I_{iU_t}^{r'} = 0$
    end if
    if $\sum_{i=1}^N I_{iU_t}^{r'} \geq \frac{N}{2}$ then
        $M_{U_t}^{r'}$ is assigned to $\mathbb{V}$
    end if
    Update $\mathbb{V}$
    [Aggregation]
    if No update from the voting layer then
        $M_G^r = M_B^r$
    else
        $M_G^r = \frac{M_G^{r-1} + \sum M_{V_q}^{r'} \times w_{V_q}^{r'}}{1 + \sum w_{V_q}^{r'}}$
        ($M_{V_q}^{r'} \in \mathbb{V}$ and $w_{V_q}^{r'}$ is calculated by a DRL model)
    end if
    return $M_G^r$
```

aggregation are shown in Fig. 2 and Algorithm 1, including five steps as follows:

Global Model Initialization: At the start of FL, the aggregator initializes the global model M_G^0 and distributes it to all participants.

Local Model Training: In epoch r ($r \geq 1$), each participant P_i trains M_G^{r-1} with its local training dataset D_i to minimize a chosen loss function by Stochastic Gradient Descent (SGD) [14]:

$$M_i^r = M_G^{r-1} - \eta \nabla \mathcal{F}(M_i^{r-1}), \tag{1}$$

where η is learning rate and the choice of the loss function $\mathcal{F}(M_i)$ depends on the model demands of different scenarios. Then, the improved model M_i^r is shared with the aggregator for the next step.

Filtering Layer: The aggregator computes distances between these shared models and selects K models with the smallest cumulative distances from others. These chosen models, denoted as $M_{S_j}^r, S_j \in \mathbb{S}$ and $\mathbb{S} \in \mathbb{N}_{\mathbb{P}}$, represent the prevailing training trend. The size of $\mathbb{S}$ is determined by the given filtering-layer coefficient K. The unselected models, denoted as $M_{U_t}^r$, $U_t \in \mathbb{U}$ and $\mathbb{U} \in \mathbb{N}_{\mathbb{P}}$, are then dropped to the voting layer. Here, the following holds:

$$\begin{cases} \mathbb{S} \cap \mathbb{U} = \emptyset, \\ \mathbb{S} \cup \mathbb{U} = \mathbb{N}_{\mathbb{P}}. \end{cases} \tag{2}$$

As the asynchronous setup, the filtering layer does not wait for the outcomes from the voting layer, allowing the aggregator to perform a preliminary pre-aggregation calculated by:

$$M_B^r = \frac{\sum_{j=1}^K \left|D_{S_j}\right| M_{S_j}^r}{\sum_{j=1}^K \left|D_{S_j}\right|}, \tag{3}$$

where, M_B^r is the result of pre-aggregation, and $|D_i|$ represents the size of the dataset D_i.

After pre-aggregation, if the voting layer, depicted as the orange box in Fig. 2, is still in progress and fails to return a new outcome to the aggregator, i.e., only the filtering outcome is updated, then FL is proceeding directly to the next epoch of learning for the sake of efficiency with the M_B^r as the final aggregation result of the current epoch, i.e.,

$$M_G^r = M_B^r, \quad \text{if only the filtering outcome is updated.} \tag{4}$$

This approach ensures that the FL process maintains momentum without delays, reducing the overall learning time cost. The M_G^r is immediately shared with the participants to start the local model learning of the next epoch.

Additionally, in each epoch, the models discarded by the filtering layer and the global model are systematically maintained in the list

$\mathscr{L}_U = \{r : \{M_{U_t}^r\}\}$ and $\mathscr{L}_G = \{r : M_G^r\}$, respectively, for the subsequent operations in the voting layer.

Voting Layer: In the voting layer, each participant P_i acts as a voter, receiving the list $\mathscr{L}_U$ and $\mathscr{L}_G$ forwarded by the aggregator. Due to the asynchronous characteristic, when the filtering layer advances to the r epoch, the voting layer handles the model dropped in the r' epoch, i.e. commences by selecting the most recent group $\{r' : \{M_{U_t}^{r'}\}\}$ from the list $\mathscr{L}_U$, where

$$r' = \max_{r_u \in \mathscr{L}_U} r_u. \tag{5}$$

Given that the voting layer commences only after the completion of the filtering layer, it follows that $r' \leq r - 1$. Moreover, we assume that the voting layer will not lag behind the filtering layer by more than e epochs, i.e., the gap between the two layers satisfies:

$$1 \leq r - r' \leq e. \tag{6}$$

The voter P_i in the voting layer casts his vote for each model dropped $M_{U_t}^{r'}$ from r' epoch according to the following and sends the voting result $I_{iU_t}^{r}$ to the aggregator:

$$I_{iU_t}^{r'} = \begin{cases} 1, & \text{if} \quad Acc_{iU_t}^{r'} \geq Acc_{iG}^{r'}, \\ 0, & \text{if} \quad Acc_{iU_t}^{r'} < Acc_{iG}^{r'}, \end{cases} \tag{7}$$

where $Acc_{iU_t}^{r'}$ and $Acc_{iB}^{r'}$ represent the accuracy of $M_{U_t}^r$ and M_B^r on the test dateset D_i^{test}, respectively.

Aggregation Finalization: Considering the possibility of Byzantine behavior and the limits on the number of malicious participants, we further set the positive voting should exceed one-half, i.e., more than half of the participants should be benign and effective voters. This setting ensures that the voting process is resilient to Byzantine attacks. Model $M_{U_t}^{r'}$ is deemed beneficial when it gains recognition from more than half of the voters and needs to be re-selected for aggregation. Hence, when the aggregator gets the voting distributed result, it evaluates the support for each voted model and denotes these re-selected models as $M_{V_q}^{r'}$ where $\mathbb{V} = \{V_q\}$ is a subset selected from $\mathbb{U}$, satisfying:

$$\begin{cases} U_t \in \mathbb{V}, & \text{if} \quad \sum_{i=1}^{N} I_{iU_t}^{r'} > \frac{N}{2}, \\ U_t \notin \mathbb{V}, & \text{if} \quad \sum_{i=1}^{N} I_{iU_t}^{r'} \leq \frac{N}{2}. \end{cases} \tag{8}$$

Next, with the assistance of (4), the aggregator updates the global model with the outcomes from the filtering and voting layers, as follows.

$$M_G^r = \begin{cases} \frac{M_B^r + \sum M_{V_q}^{r'} \times w_{V_q}^{r'}}{1 + \sum w_{V_q}^{r'}}, & \text{if the voting outcome is updated;} \\ M_B^r, & \text{if only the filtering outcome is updated,} \end{cases} \tag{9}$$

where $w_{V_q}^{r'}$ represents a weight for $M_{V_q}^r$ determined by a DRL model with information including the processing epoch number in two layers r and r', voting results $\{I_{iU_t}^{r'}\}$, and model accuracy $Acc_{iU_t}^{r'}$, $Acc_{iG}^{r'}$. This asynchronous design ensures that the global model remains efficient and effectively incorporates the most recent contributions from both layers. The updated global model M_G^r is then sent back to each participant P_i to start the local training for the next epoch.

To mitigate the adverse impact of outdated models on the global model, the voting layer is set frozen in the initial stages of FL. It only resumes to operate when the global model reaches a relatively stable state, assisting in the optimization of the global model. The duration of this frozen period can be selected based on the specific task requirements and scenarios.

Table 1
Notation definition.

Notation	Definition
N	Number of the participants in FL.
P_i	The ith participant.
D_i	The local dataset of P_i.
r	The index of a global epoch
r'	Voting layer processing epoch
M_i^r	The local model of participant P_i in r epoch.
M_G^r	The global model in r epoch.
$\mathcal{F}$	Loss function.
η	Learning rate.
$\mathbb{S}$	The set of selected models in the filtering layer.
$\mathbb{U}$	The set of dropped models into the voting layer.
$\mathbb{V}$	The set of re-selected models in the voting layer.
I_{ij}^r	The voting result of P_i for M_j^r
w_i^r	The aggregation weight for M_i in the rth epoch.

For better understanding, we summarize all symbols related to the proposed filtering-and-voting aggregation framework in Table 1.

3.3. Convergence analysis

In this section, we delve into the convergence of the proposed filtering-and-voting aggregation. Despite the asynchronous setting between the filtering and voting layers, as specified in (6), the lag between the filtering and voting layers is bounded by a predefined threshold, which is critical to preserving the convergence of the proposed framework. A comprehensive mathematical proof is provided to demonstrate that this design ensures the global model remains stable and converges in the proposed framework.

Firstly, we make the following assumptions about the loss function $\mathcal{F}$:

Assumption 1. Give a smoothness parameter $\beta > 0$, $\mathcal{F}$ is β-smooth if $\forall\ x,\ y,\ \mathcal{F}$ satisfies:

$$\mathcal{F}(y) - \mathcal{F}(x) \leq \langle \nabla \mathcal{F}(x), y - x \rangle + \frac{\beta}{2} \| y - x \|^2, \tag{10}$$

where $\langle \cdot, \cdot \rangle$ stands for inner product calculation and $\| \cdot \|$ stands for the Euclidean norm.

Assumption 2. Give a constant $\mu > 0$, $\mathcal{F}$ is μ-strongly convex if $\forall\ x$, y, $\mathcal{F}$ satisfies:

$$\mathcal{F}(y) - \mathcal{F}(x) \geq \langle \nabla \mathcal{F}(x), y - x \rangle + \frac{\mu}{2} \| y - x \|^2. \tag{11}$$

Assumption 3. There exists such an M^* that can minimize $\mathcal{F}$, i.e.,

$$M^* = \underset{M}{inf}\, \mathcal{F}(M) \quad \text{and} \quad \nabla \mathcal{F}(M^*) = 0, \tag{12}$$

where $inf(\cdot)$ stands for the infimum of a set.

These assumptions are suitable for a variety of models with convex loss functions, such as linear regression [26], logistic regression [27], and support vector machines [28], which are widely employed in practical applications. For instance, logistic regression is frequently utilized in the autonomous driving sector for image classification in vehicle vision systems [29], where it is employed to detect various objects such as pedestrians, other vehicles, and buildings to enhance navigation algorithms. Support vector machines are commonly used in text recognition tasks due to their ability to handle high-dimensional data and their effectiveness in classifying texts based on complex patterns in data features [30]. While modern machine learning models often involve non-convex loss functions, numerous tasks adhere to convex or strongly convex assumptions, such as cross-entropy [31] and least squares [32]. Models developed under these assumptions are widely employed due to their analytical tractability and practical effectiveness across various applications, even with non-convex losses.

We denote the upper bound for the gap between the local model updates and the global model updates as Q_1 and denote the upper bound for local model updates as Q_2, i.e.,

$$\begin{aligned}&\mathbb{E}\|\nabla\mathcal{F}(M_G)-\nabla\mathcal{F}(M_i)\|^2\le Q_1,\\&\mathbb{E}\|\nabla\mathcal{F}(M_i)\|^2\le Q_2,\quad\forall i\in\mathbb{N}_{\mathbb{P}}.\end{aligned}\tag{13}$$

Under these assumptions, we analyze the convergence of the proposed filtering-and-voting aggregation framework, demonstrating that the results of the voting layer do not hinder the convergence of FL. We present the convergence upper bound for the proposed filtering-and-voting aggregation framework, as established in the following theorem:

Theorem 1. *The convergence bound of the global loss function in the proposed filtering-and-voting aggregation framework after R epochs is*

$$\mathbb{E}[\mathcal{F}(M_G^R)-\mathcal{F}(M^*)]\le\frac{\beta}{2}(1-\eta\mu)^{R-e-1}\|M_G^0-M^*\|^2+Z,\tag{14}$$

where $Z=\frac{\beta}{2}\sum_{t=0}^{r-1}(1-\eta\mu)^t\eta^2Q_1+\frac{\beta}{2}\alpha(\frac{\beta\eta^2}{2}Q_2-\eta\frac{2Q_2-Q_1}{2})$.

Since $1-\eta\mu<1$, we have $\frac{\beta}{2}(1-\eta\mu)^{R-e-1}\rightarrow 0$, as $R\rightarrow\infty$. It indicates that the proposed filtering-and-voting aggregation framework can converge to a stable value, and the lag between the voting and filtering layers does not adversely affect the convergence properties.

Proof. We rewrite (9) that with the voting outcome updated part as

$$M_G^r=(1-\alpha)M_G^{r-1}+\alpha\frac{\sum w_{V_q}^{r'}M_{V_q}^{r'}}{\sum w_{V_q}^{r'}},\tag{15}$$

where $r-e\le r'\le r-1$ and $\alpha=\frac{\sum w_{V_q}^{r'}}{1+\sum w_{V_q}^{r'}}\in(0,1)$.

Similar to the proof in previous work [33], we can express M_G^r as a function of the M_G^{r-1} and $M_{V_q}^{r'}$ based on (15). Consequently, the expected difference in the loss function between M_G^r and M^* can be bounded as:

$$\begin{aligned}&\mathbb{E}[\mathcal{F}(M_G^r)-\mathcal{F}(M^*)]\\&\le(1-\alpha)\mathcal{F}(M_G^{r-1})+\alpha\mathbb{E}[\mathcal{F}(\frac{\sum w_{V_q}^{r'}M_{V_q}^{r'}}{\sum w_{V_q}^{r'}})]-\mathcal{F}(M^*)\\&\le(1-\alpha)\mathcal{F}(M_G^{r-1})+\alpha\sum\frac{w_{V_q}^{r'}}{\sum w_{V_q}^{r'}}\mathcal{F}(M_{V_q}^{r'})-\mathcal{F}(M^*)\\&\le(1-\alpha)[\mathcal{F}(M_G^{r-1})-\mathcal{F}(M^*)]+\alpha\mathbb{E}[\mathcal{F}(M_{V_q}^{r'})-\mathcal{F}(M^*)].\end{aligned}\tag{16}$$

By applying (1) and (10), the second term on the right-hand side (RHS) of the inequality is bounded by:

$$\begin{aligned}&\mathbb{E}[\mathcal{F}(M_{V_q}^{r'})-\mathcal{F}(M^*)]\\&\le\mathbb{E}[\mathcal{F}(M_G^{r'-1})-\eta\nabla\mathcal{F}(M_{V_q}^{r'-1})]-\mathcal{F}(M^*)\\&\le\mathcal{F}(M_G^{r'-1})-\eta\mathbb{E}\left\langle\nabla\mathcal{F}(M_G^{r'-1}),\nabla\mathcal{F}(M_{V_q}^{r'-1})\right\rangle\\&+\frac{\beta\eta^2}{2}\mathbb{E}\|\nabla\mathcal{F}(M_{V_q}^{r'-1})\|^2-\mathcal{F}(M^*).\end{aligned}\tag{17}$$

Using Cauchy–Schwarz inequality and (13), we have

$$\mathbb{E}\left\langle\nabla\mathcal{F}(M_G^{r'-1}),\nabla\mathcal{F}(M_{V_q}^{r'-1})\right\rangle\ge\frac{2Q_2-Q_1}{2}.\tag{18}$$

Inserting the results of (13) and (18) into (17), (17) can be rewritten as

$$\begin{aligned}&\mathbb{E}[\mathcal{F}(M_{V_q}^{r'})-\mathcal{F}(M^*)]\\&\le\mathcal{F}(M_G^{r'-1})-\mathcal{F}(M^*)-\eta\frac{2Q_2-Q_1}{2}+\frac{\beta\eta^2}{2}Q_2\\&\triangleq\mathcal{F}(M_G^{r'-1})-\mathcal{F}(M^*)+X,\end{aligned}\tag{19}$$

where $X=\frac{\beta\eta^2}{2}Q_2-\eta\frac{2Q_2-Q_1}{2}$.

Substituting (19) into (16), we have

$$\begin{aligned}\mathbb{E}[\mathcal{F}(M_G^r)-\mathcal{F}(M^*)]\le&(1-\alpha)[\mathcal{F}(M_G^{r-1})-\mathcal{F}(M^*)]\\&+\alpha[\mathcal{F}(M_G^{r'-1})-\mathcal{F}(M^*)+X].\end{aligned}\tag{20}$$

According to (10) and (12), we have

$$\begin{aligned}&\mathcal{F}(M_G^{r-1})-\mathcal{F}(M^*)\\&\le\left\langle\nabla\mathcal{F}(M^*),M_G^{r-1}-M^*\right\rangle+\frac{\beta}{2}\|M_G^{r-1}-M^*\|^2\\&\le\frac{\beta}{2}\|M_G^{r-1}-M^*\|^2,\end{aligned}\tag{21}$$

as $\nabla\mathcal{F}(M^*)=0$.

In [34], it was proved that the upper bound of the distance between M_G^{r-1} and M^* could be controlled by the distance between the previous global model and M^*, i.e.,

$$\begin{aligned}&\|M_G^{r'-1}-M^*\|^2\le(1-\eta\mu)\|M_G^{r'-2}-M^*\|^2+\eta^2Q_1\\&\le(1-\eta\mu)^{r-1}\|M_G^0-M^*\|^2+\sum_{t=0}^{r-1}(1-\eta\mu)^t\eta^2Q_1\\&\triangleq(1-\eta\mu)^{r-1}\|M_G^0-M^*\|^2+Y_1,\end{aligned}\tag{22}$$

where $Y_1=\sum_{t=0}^{r-1}(1-\eta\mu)^t\eta^2Q_1$.

Substituting (22) into (21), we have

$$\begin{aligned}&\mathcal{F}(M_G^{r-1})-\mathcal{F}(M^*)\le\frac{\beta}{2}\|M_G^{r-1}-M^*\|^2\\&\le\frac{\beta}{2}[(1-\eta\mu)^{r-1}\|M_G^0-M^*\|^2+Y_1].\end{aligned}\tag{23}$$

Similarly, we have

$$\begin{aligned}&\mathcal{F}(M_G^{r'-1})-\mathcal{F}(M^*)\le\frac{\beta}{2}\|M_G^{r'-1}-M^*\|^2\\&\le\frac{\beta}{2}[(1-\eta\mu)^{r'-1}\|M_G^0-M^*\|^2+\sum_{t=0}^{r'-1}(1-\eta\mu)^t\eta^2Q_1]\\&\triangleq\frac{\beta}{2}[(1-\eta\mu)^{r'-1}\|M_G^0-M^*\|^2+Y_2],\end{aligned}\tag{24}$$

where $Y_2=\sum_{t=0}^{r'-1}(1-\eta\mu)^t\eta^2Q_1$.

By substituting (23) and (24) into (20), the convergence bound can be rewritten as

$$\begin{aligned}&\mathbb{E}[\mathcal{F}(M_G^r)-\mathcal{F}(M^*)]\\&\le\frac{\beta}{2}(1-\alpha)[(1-\eta\mu)^{r-1}\|M_G^0-M^*\|^2+Y_1]\\&+\frac{\beta}{2}\alpha[(1-\eta\mu)^{r'-1}\|M_G^0-M^*\|^2+Y_2+X]\\&=\frac{\beta}{2}[(1-\alpha)(1-\eta\mu)^{r-1}+\alpha(1-\eta\mu)^{r'-1}]\|M_G^0-M^*\|^2\\&+\frac{\beta}{2}(1-\alpha)Y_1+\frac{\beta}{2}\alpha(Y_2+X).\end{aligned}\tag{25}$$

As $r-e\le r'\le r-1$, we have $Y_2\le Y_1$, and the following inequality holds:

$$\begin{aligned}&(1-\alpha)(1-\eta\mu)^{r-1}+\alpha(1-\eta\mu)^{r'-1}\\&\le(1-\eta\mu)^{r'-1}\\&\le(1-\eta\mu)^{r-e-1}.\end{aligned}\tag{26}$$

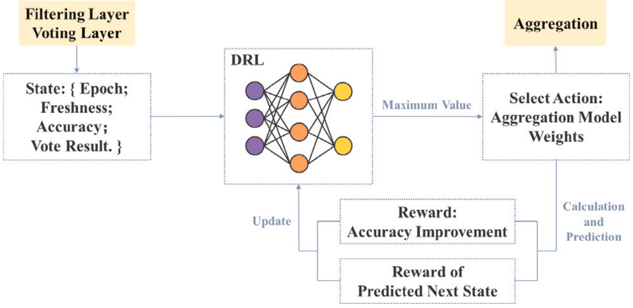

Filtering Layer
Voting Layer
Aggregation
DRL
State: { Epoch; Freshness; Accuracy; Vote Result. }
Maximum Value
Select Action: Aggregation Model Weights
Update
Reward: Accuracy Improvement
Reward of Predicted Next State
Calculation and Prediction

Fig. 3. DRL-based dynamic aggregation weight selection with the roles of the agent (aggregator), the environment (FL aggregation process), states (intermediate results from the filtering layer and the voting layer), and actions (weight selections for the target model).

So, (25) can be rewritten as

$$\begin{aligned}&\mathbb{E}[\mathcal{F}(M_G^r) - \mathcal{F}(M^*)]\\&\leq \frac{\beta}{2} \leq (1-\eta\mu)^{r-e-1}\|M_G^0 - M^*\|^2 + \frac{\beta}{2}Y_1 + \frac{\beta}{2}\alpha X\\&\triangleq \frac{\beta}{2} \leq (1-\eta\mu)^{r-e-1}\|M_G^0 - M^*\|^2 + Z,\end{aligned} \tag{27}$$

where $Z = \frac{\beta}{2}Y_1 + \frac{\beta}{2}\alpha X = \frac{\beta}{2}\sum_{t=0}^{r-1}(1-\eta\mu)^t\eta^2 Q_1 + \frac{\beta}{2}\alpha(\frac{\beta\eta^2}{2}Q_2 - \eta\frac{2Q_2-Q_1}{2})$. □

Theorem 1 confirms that the inclusion of a voting layer does not impede the execution of FL tasks. This assurance of convergence validates the practicality and underscores its effectiveness in FL environments, reinforcing the notion that integrating additional layers, such as the voting layer, enhances the learning process. The convergence upper bound provided in Theorem 1 demonstrates that the proposed filtering-and-voting aggregation framework is convergent. Our convergence analysis is closely related to practical applications, which we further substantiate in Section 5 through experiments on image classification.

4. DRL-based adaptive aggregation weight selection

A DRL model is introduced in the proposed frameworks for dynamic aggregation weight selection to better leverage the value of models selected by the voting layer. This section elaborates on the elements of the DRL model used for dynamic aggregation weight selection in the proposed filtering-and-voting aggregation framework, as shown in Fig. 3.

Different DRL variables, parameter spaces (discrete or continuous), and DRL algorithms such as Deep Q-network (DQN) [35], and Trust Region Policy Optimization (TRPO) [36], can be chosen based on the specific FL task and device capabilities, providing a flexible and scalable framework implementation. Additionally, the DRL model can be trained synchronously during FL or pre-trained. Both online training DRL model and pre-trained DRL models in FL can effectively enhance the performance and security of the global model.

The composition of the DRL model used in our work consists of the following elements:

Agent. The agent is the decision-maker who executes the learning process and interacts with the environment. Here, the aggregator is an agent in the DRL for filtering-and-voting aggregation weight selection.

Environment. The environment means the external system with which the agent interacts. The environment is typically modeled to represent the problem space or the scenario in which the agent operates. It responds to the actions of the agent and presents new states to the agent. In this paper, the FL aggregation process is the environment.

States. A representation of the current environmental situation is called the state of the DRL. States capture relevant information that the agent needs to make decisions.

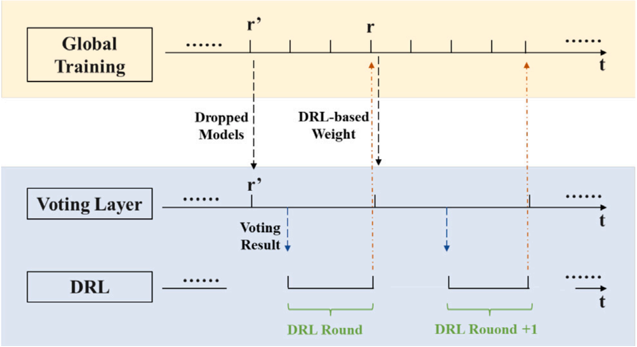


Fig. 4. The timing and dependency relationship between the DRL-based aggregation weight selection and the voting layer within the proposed FL framework. Each round of DRL is triggered by the result of a voting layer epoch, depicted sequentially but not continuously. The voting layer processing epoch is set as continuous training and is delayed relative to the global FL epochs.

In this paper, the agent (the aggregator) obtains the intermediate results of the filtering layer and voting layer as the observation of the environment, which is fed into our DRL model as input. Specifically, the intermediate results include the following:

- Current learning epoch of FL, calculated based on the number of rounds conducted by the filtering layer, i.e., r.
- Freshness of the voting layer, which indicates the degree to which the voting layer lags behind the filtering layer, i.e., r'/r.
- Accuracy of the target model, for whom the weight is calculated by this DRL model, $Acc(M_{target})$. It is the most direct indicator of the contribution of the target model to the global model.
- Voting outcomes from voters regarding that model, i.e., the proportion of voters who believe that the inclusion of the target model would benefit the global model: $\sum I_{iU_t}^{r'}/N$. This ratio represents a critical metric for assessing consensus and the perceived value of individual model contributions within FL.

Actions. The set of all possible decisions the agent can make in a given state. The choice of action affects the environment, leading to a new state. Here, the weights that can be chosen for the target model are actions, denoted as W. The weights can be either discrete or continuous. In our experiments, some discrete weights are used as actions to simplify the problem and enhance efficiency. The action selected for the target model M_q in r epoch is denoted as w_q^r.

Reward Function. A function that assigns a numerical value to each action as feedback to the agent about the effectiveness of its decision. The objective of the agent in DRL is to maximize the reward, which guides the learning of an optimal policy. In this paper, the reward function calculates the contribution of the target model aggregated with different weights to the global model, i.e., the model accuracy improvement for each weight:

$$\begin{aligned}R(w, M_{target}) =& Acc((1-w)\times M_G + w \times M_{target})\\&- Acc(M_G),\end{aligned} \tag{28}$$

where the w is the selected aggregation weight (i.e., action) of the target model M_{target}.

Note that the DRL-based aggregation weight selection relies on the outcomes from the voting layer. Each round of DRL corresponds directly to an epoch of the voting layer, as shown in Fig. 4. From a temporal perspective, the round of DRL is not continuous because the start of the DRL must wait for the voting results. As for the voting layer, we designate the end of a DRL round as the conclusion of a voting layer processing epoch and the determination of the new r' according to (5) as the start of the next epoch. It makes the voting layer processing epochs sequential but chronologically behind the global FL epoch. This temporal alignment keeps the voting process continuously updated, yet systematically delayed relative to the global FL training epoch, allowing for effective training without compromising the integrity of the training process.

5. Experiment

In this section, we experimentally validate the performance of the proposed filtering-and-voting aggregation framework in FL. The results underscore the adaptability of the proposed framework in scenarios where FL involves both special participants with unique training data and malicious participants launching poisoning attacks. Our findings illustrate how the framework adeptly assesses the value of shared models from different participants, effectively safeguarding against attacks from malicious participants while maximizing the utilization of valuable information contributed by the participants with unique data to enhance the overall performance of the global model.

5.1. Experimental setting

We consider an FL scenario with 10 participants, including one special participant with unique data and two malicious participants, alongside seven general participants. Datasets utilized for FL are divided into two categories: one consists of rare and distinct data, which is accessible exclusively to the special participant; the other category contains common data, which is owned by all participants. To emphasize the non-iid nature of the local dataset held by the participants, the quantity of different common label data varies among participants. Each participant allocates a portion of their data as a training set for model learning and another portion as a testing dataset for accuracy testing. Additionally, participants supplement their testing datasets with data randomly sampled from a small-sized third-party dataset with all labels. This setup mirrors real-world scenarios, such as in fault diagnosis tasks, where participants may not own fault data for training purposes but can utilize such data for testing. The malicious participants conduct poisoning attacks in the local model training process and engage in Byzantine behavior in the voting layer, randomly voting “1” or “0” with a given probability.

The FL process is set with 500 training epochs, and the voting layer in the proposed framework starts from the 100th epoch. We use a Convolutional Neural Network (CNN) [37] for the FL image recognition task. The initial model is distributed to all participants by the aggregator, and the CNN parameters are set with reference to LeNet-5 model [38]. The proposed framework is compatible with prevalent DRL models, and existing work [39] demonstrates the efficiency and feasibility of DQN-based weight selection in FL under resource-constrained conditions. To reduce experimental costs, we employ a typical DQN model [40] for the dynamic selection of aggregation weight. The input variables of the DQN model include FL epoch, freshness of the voting layer, model accuracy and voting results. The output of the DQN model is chosen from a discrete parameter space. Our experiments are run on NVIDIA PCIe A100, 2 × 40 GB, and we use Python 3.7.16 and TensorFlow 2.11.0 to build and train the proposed FL framework.

5.2. Benchmarks and experimental groups

We choose FedAvg [14], multi-Krum [5], and FLAME [16] as benchmarks for the control group due to their widespread recognition and application in FL. FedAvg [14] is the first and classic FL aggregation algorithm, which maximizes the use of shared information from all participants in scenarios absent of malicious participants. FedAvg is an inevitable benchmark when discussing FL aggregation algorithms. Another extensively discussed FL aggregation algorithm is multi-Krum [5], which introduces a selective aggregation algorithm based on model distances and achieves BFT in the presence of malicious participants. Since its presentation, multi-Krum and its variants have remained popular in FL applications [41]. Many recent studies about FL secure aggregation still revolve around multi-Krum, combining with homomorphic encryption and multi-party secure computation techniques [42] or clustering algorithms [43,44]. Given the centrality and importance of multi-Krum in these approaches, we choose it as another benchmark for our experiments.

Furthermore, we consider a cosine-similarity-based algorithm, FLAME [16], as a state-of-the-art representative of secure FL aggregation. FLAME clusters the models shared by participants to select those that align with the majority, thereby resisting the intrusion of malicious models. Additional norm-based median clipping and noise addition enhance the robustness and generalization capability of FLAME. Through comparisons with these typical methods, we aim to highlight the distinctive security and performance features of our filtering-and-voting aggregation framework.

As for the experimental groups, we establish three sets of filtering-and-voting aggregation approaches with different methods of selecting aggregation weights. The training of the DRL model used in the proposed framework can either be training during the FL process or conducted beforehand. Therefore, we demonstrate the performance of both the proposed framework with online DRL and the proposed framework with pre-trained DRL. To illustrate the advantages of DRL-based dynamic aggregation weight selection, we also consider a filtering-and-voting framework with a fixed aggregation weight. Its process is essentially the same as depicted in Fig. 2, with the only difference being that during the final aggregation, a fixed aggregation weight is employed to combine the models endorsed by the voting layer and those selected by the filtering layer.

In our case, we choose the parameter $K = 5$ in the filtering layer, the same as the commonly used parameter in multi-Krum. For the filtering-and-voting framework with a fixed aggregation weight, we employ $1/6$ as the given aggregation weight to represent an average aggregation with the K models selected by the filtering layer. The parameters used in FLAME are configured consistently with those in [16].

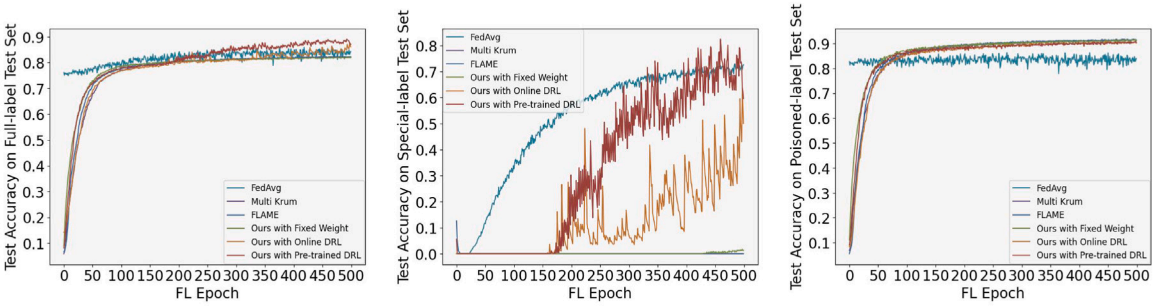
5.3. Performance across different datasets

To demonstrate the performance of the proposed framework across different datasets, we use the MNIST [45] and CIFAR-10 [46] datasets, which are widely utilized in the fields of computer vision and neural networks. The MNIST dataset comprises labeled hand-written digit images from 0 to 9, where images labeled as ‘0’ are designated as rare and distinct data accessible only to the special participant. The remaining labels are randomly assigned to all participants. The CIFAR-10 dataset features images categorized under various labels, including airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck. In our experiments, the ‘airplane’ category is assigned as unique data, whereas the other labels are considered common data. This differentiation in data access among participants simulates the FL scenario with imbalanced training data.

The malicious participants here conduct label-flipping poisoning attacks [17], altering the labels of all training data to the same value (designated ‘9’ in the MNIST dataset and ‘truck’ in the CIFAR-10 dataset) to compromise the performance of the global model. To show the aggregation strategies of the proposed framework, we evaluate the global model’s test accuracy across various test datasets, i.e. full-label est set, special-label test set, and poisoned-label test set. Additionally, we provide insights into the differential selection of aggregation weights by the proposed framework for each type of participant- general participant, special participant with unique data, and malicious participant with poisoned data.

Table 2
Average global model accuracy on various test sets.

Dataset	Aggregation method		Test accuracy on full-label data	Test accuracy on special-label data	Test accuracy on poisoned-label data
MNIST	FedAvg [14]		81.05% ± 2.11%	69.59% ± 1.12%	80.22% ± 2.53%
	Multi-Krum [5]		82.07% ± 0.03%	0.00% ± 0.00%	89.19% ± 0.05%
	FLAME [16]		82.25% ± 0.05%	0.00% ± 0.00%	89.59% ± 0.15%
	Ours with	Fixed weight	82.90% ± 0.15%	2.67% ± 2.67%	89.70% ± 0.28%
		Online DRL	**87.24% ± 1.58%**	**66.28% ± 2.60%**	**90.79% ± 0.09%**
		Pre-trained DRL	**89.03% ± 0.07%**	**73.67% ± 1.33%**	**90.60% ± 0.09%**
CIFAR-10	FedAvg [14]		52.42% ± 2.19%	44.37% ± 3.63%	51.15% ± 3.99%
	Multi-Krum [5]		57.52% ± 1.17%	0.00% ± 0.00%	65.79% ± 1.55%
	FLAME [16]		59.35% ± 1.72%	0.00% ± 0.00%	66.83% ± 1.96%
	Ours with	Fixed weight	60.91% ± 1.26%	1.83% ± 1.83%	69.05% ± 1.03%
		Online DRL	**65.07% ± 1.78%**	**46.99% ± 0.76%**	**70.70% ± 0.82%**
		Pre-trained DRL	**65.71% ± 1.95%**	**49.09% ± 1.03%**	**71.29% ± 1.16%**



(a) Accuracy on Full-label Test Set (b) Accuracy on Special-label Test Set (c) Accuracy on Poisoned-label Test Set

Fig. 5. Case study of the global model test accuracy development throughout the FL training process (using the MNIST dataset). As the voting layer in all filtering-and-voting frameworks starts from the 100th epoch and changes take time to materialize, the advantage of filtering-and-voting frameworks shows after the 150th epoch, especially those utilizing DRL. The proposed filtering-and-voting framework with DRL comes evident with a marked increase in test accuracy, outperforming traditional methods like FedAvg [14] and multi-Krum [5] and state-of-the-art methods like FLAME [16].

5.3.1. *Test accuracy*

As shown in Table 2, the average accuracies of the global models across different datasets from the proposed methods and benchmarks reflect the superior performance of the proposed framework. Whether on the MNIST or CIFAR-10 datasets, both the proposed framework with pre-trained DRL and the proposed framework with online DRL show significant performance advantages with consistently higher accuracy, which highlights the effectiveness and adaptability of the proposed frameworks in handling different types of data challenges.

On the full-label test set, the accuracy of the proposed framework with pre-trained DRL is the highest, surpassing FedAvg by 7.98% with the MNIST dataset and 13.29% with the CIFAR-10 dataset; surpassing multi-Krum by 6.96% with the MNIST dataset and 8.19% with the CIFAR-10 dataset; and surpassing FLAME by 6.78% with the MNIST dataset and 6.36% with the CIFAR-10 dataset. The accuracy of the proposed framework with online DRL is second and significantly higher than that of the other control groups. Due to CIFAR-10's color composition and greater complexity, the overall accuracy of the CIFAR-10 dataset is consistently lower than the accuracy of the MNIST dataset.

On the special-label test set, whether using the MNIST or CIFAR-10 datasets, multi-Krum and FLAME completely neglect the shared model from special participants with unique data, and the fixed-parameter filtering-and-voting framework uses too little information from special participants with unique data, resulting in their loss of recognition capability on the special label. The accuracy of the proposed framework with pre-trained DRL remains the highest, outperforming FedAvg, while the performance of the proposed framework with online DRL is slightly inferior to FedAvg but still comparable. It demonstrates that the proposed framework effectively leverages the information from special participants with unique data to optimize the performance of the global model. On the poisoned-label test set, FedAvg performs poorly because no protective measures are in place, whereas the other methods are good at identifying and resisting malicious participants.

Specifically, Fig. 5 shows the development of the global model's test accuracy during a specific FL training process for various aggregation methods. The case study uses the MNIST dataset, and similar development trends are observable when employing the CIFAR-10 dataset. It is worth noting that the voting layer in all filtering-and-voting frameworks starts from the 100th epoch. Thus, before its work, these frameworks are similar to multi-Krum. However, the advantages of the proposed frameworks are not immediately apparent from the start of the voting layer's operation. The information from special participants needs time to accumulate to truly impact the performance of the global model and be reflected in the test accuracy.

As shown in Fig. 5, starting from the 150th epoch, the optimization effect of the voting layer on the global model becomes apparent, and both DRL-based filtering-and-voting frameworks begin to demonstrate higher accuracy than multi-Krum and FLAME, especially on the special-label test set. The fixed-parameter method fails to markedly surpass multi-Krum because it does not utilize the information related to special labels as extensively. In essence, this method requires a larger number of epochs to accumulate sufficient information from special participants to enhance the recognition capabilities for the special label. The proposed frameworks with online DRL and pre-trained DRL achieve accuracy on the special label that approaches or even surpasses FedAvg, as shown in Fig. 5(b). This highlights that the proposed frameworks effectively utilize information from special participants with unique data and enhance the performance of the global model by preserving the information from diverse and benign shared models. Compared to our framework with pre-trained DRL, the framework with online

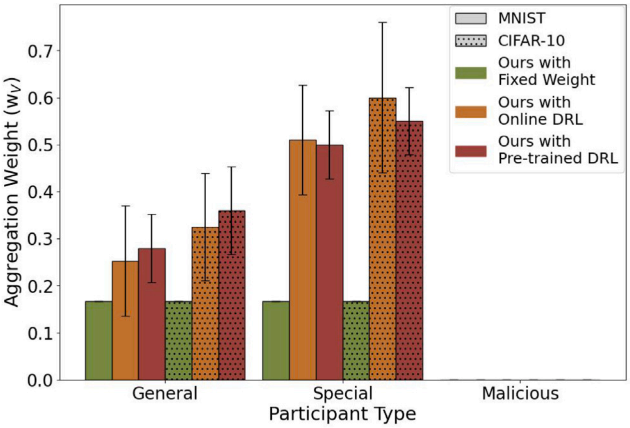


Fig. 6. Aggregation weight for different participant types selected by our methods. As the proposed double-layer frameworks effectively exclude models provided by the malicious participant, their aggregation weights remain zero. Aggregation weight selection via pre-trained DRL exhibits more stability and achieves a similar mean as the DRL that keeps training during the FL process. Due to the greater variability among the CIFAR-10 dataset, amplifying the aggregation weights of benign participants, particularly those with unique data, enhances the accuracy of the global model. Consequently, the proposed methods tend to assign larger aggregation weights to both general and special participants for the CIFAR-10 dataset than for the MNIST dataset, aiming to better harness the diverse characteristics of the dataset.

DRL requires additional computational resources to train the DRL model, thereby necessitating more time to manifest its improvements in accuracy.

5.3.2. Aggregation weight selection

Through the comparison and analysis of the global model accuracy, we note the significance of aggregate parameter selection. Considering that the filtering-and-voting framework with fixed aggregation weights performs significantly worse than methods employing dynamic weight selection, DRL-based dynamic aggregate parameter selection enables the maximization of the voting layer's impact, ensuring that each model endorsed by voters receives the most suitable parameters for combination with the models selected by the filtering layer.

Fig. 6 illustrates the mean and variance of aggregation weights (i.e., w_{V_q} in (9)) assigned to shared models from different participants in the filtering-and-voting framework. Whether using the MNIST or CIFAR-10 datasets, the proposed filtering-and-voting framework effectively excludes models provided by the malicious participant through the voting layer, by setting their aggregation weights to zero, regardless of the aggregation weight selection methods. Fixed weight here is set at $1/6$ to facilitate the averaging aggregation with K models selected by the filtering layer, embodying the core concept of FedAvg. This setting is a typical example of aggregation weight configuration commonly used in a FL environment. Compared to the fixed weight, the aggregation weights derived from the proposed DRL-based methods are generally larger, expanding the influence of models approved by the voters. Aggregation weight selection via pre-trained DRL exhibits stronger stability, with lower variance, yet achieves a similar mean value across multiple experiments as the online DRL that keeps training during the FL process. Combined with the comparison of the test accuracy in Table 2, it suggests that higher aggregation weights for voter-approved models can effectively enhance model performance and generalization capability.

Models shared by special participants with unique data are assigned higher aggregation weights compared to those shared by general participants, highlighting that special participants with unique data have a more significant positive impact on the performance of the global model. This shows the significance of the proposed framework, particularly in scenarios where the local training datasets of FL participants are non-iid. Important yet scarce data information can be recognized by the voting layer and effectively amplified through the RL-based aggregation weight selection method. The proposed framework allows the global model to achieve valuable enhancements by capitalizing on diverse data contributions to improve overall model efficacy.

By comparing the aggregation weight selection across different datasets, it can be observed that the proposed DRL-based methods tend to allocate larger weights to both general and special participants in the CIFAR-10 dataset than in the MNIST dataset. This is attributed to the substantial variability within the CIFAR-10 dataset, characterized by its complex and diverse data. In such contexts, the importance of aggregation diversity is further emphasized. Amplifying the aggregation weights of benign participants, particularly those with unique datasets, is crucial for enhancing the accuracy of the global model by capitalizing on their distinct contributions. This contrast underscores the effectiveness of the proposed DRL-based aggregation weight selection, which can adeptly utilize data diversity to optimize the performance and accuracy of the global model in scenarios marked by diverse and complex data types.

The DRL-based dynamic aggregation weight is indispensable for the proposed filtering-and-voting aggregation framework. Through the selection of aggregation weight, the limitations of average-based aggregation methods can be overcome, maximizing the utilization of rare and valuable information related to specific labels in the case of non-iid training data. This optimization process enhances the recognition accuracy and generalization ability of the global model.

5.4. Performance under different attacks

In this paper, we explore an open FL scenario, where the shared model parameters are accessible to all participants; thus, sending dropped models from the filtering layer to all voters does not introduce additional risks. However, in this context, attackers are omniscient and aware of all other participants' shared model parameters, enabling them to execute more sophisticated and targeted attacks. So we conduct experiments with the MNIST dataset to evaluate the performance of the proposed framework under various FL poisoning attacks, including the classical label-flipping attack [17] used in the previous experiments and three advanced state-of-the-art poisoning attacks.

The label-flipping attack [17], manipulates model accuracy by corrupting genuine labels into incorrect ones, thereby introducing tainted data. In FL, a label-flipping attack only requires an attacker to modify their local data and train with the altered, incorrect data without requiring knowledge of the models shared by other benign participants or the aggregation strategy employed by the aggregator. Consequently, while label-flipping attacks are straightforward to execute, they are also relatively easy to mitigate with common FL secure aggregation methods, such as multi-Krum, which can effectively defend against such simplistic adversarial interventions. Here we maintain consistency with previous experiments regarding the setup for the label-flipping attack by altering all training data labels to a uniform value—specifically, labels are changed to '9' in the MNIST dataset and to 'truck' in the CIFAR-10 dataset.

When attackers gain access to all benign shared models, they can leverage this information to further refine their attack models, making them more difficult to be detected and defended against. This access allows attackers to understand the characteristics of the benign models, enabling them to craft attacks that blend more seamlessly with normal activities, thus posing a significant challenge to existing defense mechanisms. The ALIE attack [47] utilizes the mean and variance of benign models to determine a perturbation range for malicious model parameters, ensuring that the adversarial modifications do not significantly deviate from those of benign models, thereby increasing the likelihood of a successful attack. In our experiments, we followed the method outlined in [47] to calculate the perturbation range and craft malicious models, with the perturbation factor set to $z = 0.1$ as per the

Table 3
Average global model accuracy on full-label test sets under various attacks.

Aggregation method		Label-flipping attack [17]	ALIE attack [47]	Min-Max attack [48]	Fang attack [7]
FedAvg [14]		81.05% ± 2.11%	89.37% ± 0.67%	89.41% ± 0.19%	85.94% ± 0.09%
Multi-Krum [5]		82.07% ± 0.03%	82.12% ± 0.11%	81.84% ± 0.13%	81.87% ± 0.09%
FLAME [16]		82.25% ± 0.05%	82.48% ± 0.07%	82.55% ± 0.07%	82.06% ± 0.18%
Ours with	Fixed weight	82.90% ± 0.15%	83.84% ± 0.31%	84.30% ± 0.22%	83.65% ± 0.18%
	Online DRL	**87.24%** ± **1.58%**	**89.26%** ± **0.71%**	**89.86%** ± **0.07%**	**86.13%** ± **0.11%**
	Pre-trained DRL	**89.03%** ± **0.07%**	**89.87%** ± **0.65%**	**90.16%** ± **0.13%**	**86.42%** ± **0.21%**

Table 4
Average percentage of different types of participants selected for aggregation under various attacks.

Aggregation method		Label-flipping attack [17]			ALIE attack [47]			Min-Max attack [48]			Fang attack [7]		
		General	Special	Malicious	General	Special	Malicious	General	Special	Malicious	General	Special	Malicious
FedAvg [14]		100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Multi-Krum [5]		71.43%	0.00%	0.00%	65.29%	0.00%	21.50%	66.57%	0.00%	17.01%	61.51%	0.00%	34.70%
FLAME [16]		86.86%	0.00%	0.00%	82.03%	0.00%	18.50%	80.29%	0.00%	29.26%	78.31%	0.00%	30.10%
Ours with	Fixed weight	78.86%	74.52%	0.00%	78.07%	71.00%	10.26%	86.63%	57.21%	13.61%	82.71%	49.03%	17.35%
	Online DRL	81.86%	86.77%	0.00%	80.86%	76.02%	9.07%	89.29%	65.40%	11.03%	85.83%	55.03%	16.60%
	Pre-trained DRL	85.11%	98.91%	0.00%	81.57%	76.04%	8.50%	90.64%	73.51%	10.30%	87.57%	64.02%	14.57%

experimental details in [47]. Unlike the fixed perturbation range in the ALIE attack, the Min-Max attack [48] introduces a dynamic selection process for the perturbation factor, aiming to maximize the deviation caused by the malicious model while ensuring that the maximum distance between the malicious model and benign models remains less than the maximum distance among benign models. The experimental setup for parameters followed the configurations in [48], starting with a perturbation factor $\gamma_{init} = 0.01$ and a change threshold $\tau = 10^{-5}$.

Expanding the capabilities of attackers to an omniscient level, where they are aware of all benign shared models and the aggregation strategy, allows them to tailor their adversarial models specifically to exploit the aggregation mechanism. Represented by Fang's attack [7], such attackers can simulate the aggregator's process to ensure their malicious models are incorporated while maximizing the disruption caused. Based on the upper bound of the perturbation factor provided in [7], we determined the maximum perturbation factor for our experiments as $\lambda_{max} = 0.001$. Following the experimental setup in [7], we also established the minimum perturbation factor as $\lambda_{min} = 10^{-5}$, allowing for finely tuned adjustments to enhance the impact of the attack within controlled parameters.

To evaluate the defensive capabilities of the proposed framework against various attacks, malicious participants are instructed to implement these four different poisoning attack techniques within the FL setting. Special participants possess unique and rare data (label 0), while other data is randomly distributed among all participants as their local datasets. We assessed the defensive capabilities of different aggregation methods by comparing their accuracy on full-label test sets. Additionally, we discussed the selection of participants for aggregation under various attacks by different methods, highlighting how the proposed framework optimizes the performance of the global model by enhancing the diversity and security of the aggregation process.

5.4.1. *Defensive capability*

As shown in Table 3, the average accuracy of the global model under various attacks demonstrates the superiority of the proposed method compared to other FL security aggregation methods, such as multi-Krum and FLAME. The proposed framework with pre-trained DRL consistently achieves the highest accuracy across all attack scenarios, including label-flipping, ALIE, Min-Max, and Fang's attacks. It outperforms multi-Krum by an average of approximately 7.5% and FLAME by approximately 7%.

Unlike malicious models in ALIE, Min-Max, and Fang's attacks, which aim to mimic benign models, those deployed in the label-flipping attack significantly deviate from benign behaviors, causing greater detrimental impacts on the global model. Consequently, FedAvg aggregates malicious models and performs worst under label-flipping attacks. However, in targeted attacks like ALIE, Min-Max, and Fang's attacks, FedAvg outperforms multi-Krum and FLAME. This is because the malicious models, specifically tailored to deceive the selective aggregation methods of multi-Krum and FLAME, shift the focus towards malicious contributions, while FedAvg's broader inclusion of benign models dilutes the malicious perturbations [47]. Meanwhile, due to the non-iid distribution of training data among participants in FL, and the fact that special label data is exclusively held by special participants, the accuracy of methods like multi-Krum and FLAME is further impeded. These methods often exclude special participants from aggregation, hence missing out on leveraging their unique contributions, which are crucial for improving the model's performance on rare but significant data classes.

By integrating as many benign models as possible, the proposed filtering-and-voting framework achieves performance on par with or even superior to FedAvg under ALIE, Min-Max, and Fang's attacks. During label-flipping attacks, the weaknesses of FedAvg's unconditional aggregation are avoided by implementing a double-layer validation that ensures the security of the model. Experimental results across various attacks demonstrate the effectiveness of the filtering and voting framework in leveraging benign contributions while mitigating the impact of malicious inputs.

5.4.2. *Aggregation participants selection*

To demonstrate the effective selection and re-aggregation of beneficial models from different participants in the proposed filtering-and-voting framework, we calculate the average percentage of participant models used for aggregation under various attacks. Since the selection of shared models for aggregation is only relevant to the filtering-and-voting architecture, the impact of the aggregation weight selection methods is not markedly pronounced. As shown in Table 4, compared with FedAvg, multi-Krum, and FLAME, whether employing fixed weight, online DRL, or pre-trained DRL methods in the proposed framework, the outcomes regarding the selection of aggregation models exhibit a degree of similarity across various scenarios.

Under the label-flipping attack, the proposed framework can identify and discard the model shared by the malicious participant with 0% aggregation probability, equal to multi-Krum and FLAME, which showcases its robust defense against data poisoning attacks. Simultaneously, the proposed framework exhibits an around 90% probability of recognizing special participants with unique data and incorporating its information into the aggregation process, which is similar to the utilization of special information in FedAvg. This finding highlights the ability of the proposed framework to discern different participant roles effectively, consistent with the presented objective demonstrated

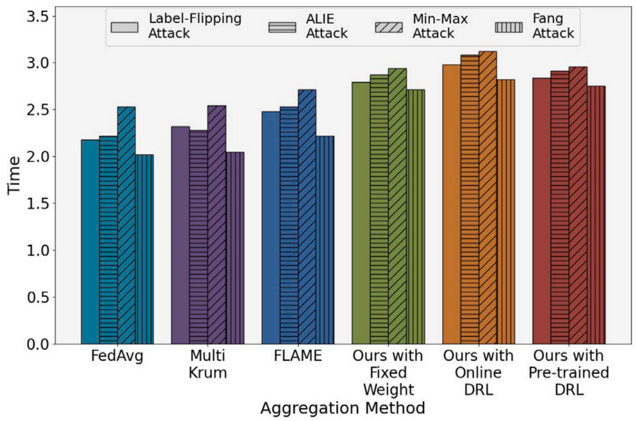


Fig. 7. Average training time per epoch for FL with different aggregation methods under various attacks. The proposed framework requires slightly more time per FL epoch, compared to FedAvg, multi-Krum, and FLAM, yet remains around 3 s per epoch. Given the significant performance improvements achieved by our methods, this slight increase in time cost is acceptable.

in Fig. 1, selecting the optimal aggregation for the global model while resisting malicious attacks, even in FL scenarios with non-iid training data.

Under advanced poisoning attacks, such as ALIE, Min-Max, and Fang's attacks, the challenge of identifying malicious models is significantly heightened due to their sophisticated design, which makes them closely resemble benign models. Despite this, the proposed framework manages to confine the impact of these malicious models on the global model to a controllable extent. It integrates only about 10% of such malicious models into the aggregation, effectively halving the likelihood compared to classical secure aggregation methods like multi-Krum and state-of-the-art approaches like FLAME. Additionally, compared to multi-Krum and FLAME, the proposed framework incorporates more benign models into the aggregation process, particularly those shared by special participants. This strategy not only dilutes the potential poisoning impact from malicious participants, thereby enhancing the robustness of the global model, but also significantly increases the diversity of the aggregation in non-iid data scenarios, which boosts the accuracy and generalization capability of the global model.

5.4.3. *Efficiency*

Fig. 7 presents the average time consumed by different aggregation methods per epoch of FL training under various attacks. In the proposed filtering-and-voting framework, the voting layer is set to start from the 100th epoch, prompting us to calculate the time from the start of the voting layer.

As seen in Fig. 7, the proposed framework requires slightly more time per FL epoch, compared to FedAvg, multi-Krum, and FLAM, yet remains around 3 s per epoch. In the proposed filtering-and-voting framework, the filtering layer can proceed to the next FL epoch without awaiting the outcome of the voting layer, thus adding only a minimal time cost. Given the significant performance improvements achieved by our methods, this slight increase in time cost is acceptable.

During Fang's attack, where malicious models simply require the perturbation of benign models without local training, and the perturbation factor is straightforward to compute, all methods exhibit a notably shorter average time per epoch of FL training. Conversely, the Min-Max attack involves a complex process to find the optimal perturbation factor, consequently demanding the most time among the attacks evaluated.

The filtering-and-voting aggregation framework utilizing DRL-based dynamic aggregation weight selection results in longer processing times compared to the fixed-weight method due to the added computational demands of DRL. However, employing pre-trained DRL is more time-efficient than online DRL concurrently with the FL process. Overall, the proposed filtering-and-voting aggregation framework distinguishes special and malicious participants with a relatively minor time cost, leading to notable enhancements in security and accuracy.

6. Conclusion and future work

In this paper, we introduced an innovative filtering-and-voting aggregation framework for FL, specifically designed to address the challenges posed by non-iid data and the potential threats of adversarial attacks. Combined with a DRL-based method for dynamic aggregation weight selection, the proposed filtering-and-voting framework enhanced the performance of the global model by effectively incorporating valuable contributions from all participants, including those with rare or specific data, while maintaining strong defenses against malicious activities. Our experimental findings revealed a significant enhancement in model accuracy and security, underscoring the efficacy of the proposed framework in optimizing information utilization in FL environments. The introduction of a filtering-and-voting aggregation process, coupled with the strategic application of DRL for weight optimization, offered a robust solution to distributed model training.

In this paper, we focused on secure and diverse aggregation in open FL scenarios, with potential future applications to practical fields, such as energy networks, healthcare data analysis, and smart city infrastructure. Considering the privacy challenges of model parameter leakage, our future work will focus on exploring voting mechanisms that do not disclose shared model details. In the proposed framework, the voting process relies on the accuracy of the voted model on the voters' local test sets, rather than on the precise model parameters themselves. Therefore, integrating Homomorphic Encryption (HE) [49] and Secure Multi-Party Computation (MPC) [50] emerges as a viable strategy to mitigate the risk of parameter leakage. By employing HE or MPC, the original model parameters can be encrypted or secured versions, effectively reducing the risks associated with model parameter leakage in FL. Additionally, integrating Differential Privacy (DP) [51] techniques can obscure the features of the models shared by participants and their local training data to further mitigate inference attacks and enhance the privacy protection of the FL system.

CRediT authorship contribution statement

Yanna Jiang: Writing – original draft, Methodology, Formal analysis, Conceptualization. **Baihe Ma:** Writing – original draft, Methodology. **Xu Wang:** Writing – review & editing, Conceptualization. **Guangsheng Yu:** Writing – review & editing, Methodology, Conceptualization. **Caijun Sun:** Visualization, Validation, Resources, Data curation. **Wei Ni:** Writing – review & editing, Supervision. **Ren Ping Liu:** 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.

Data availability

Data will be made available on request.

References

[1] T. Li, A.K. Sahu, A. Talwalkar, V. Smith, Federated learning: Challenges, methods, and future directions, IEEE Signal Process. Mag. 37 (3) (2020) 50–60.

[2] J.-P.A. Yaacoub, H.N. Noura, O. Salman, Security of federated learning with IoT systems: Issues, limitations, challenges, and solutions, Internet Things Cyber-Phys. Syst. 3 (2023) 155–179.

[3] Y. Jiang, B. Ma, X. Wang, G. Yu, P. Yu, Z. Wang, W. Ni, R.P. Liu, Blockchained federated learning for internet of things: A comprehensive survey, ACM Comput. Surv. 56 (10) (2024).

[4] P. Kairouz, H.B. McMahan, B. Avent, A. Bellet, M. Bennis, A.N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al., Advances and open problems in federated learning, Found. Trends® Mach. Learn. 14 (1–2) (2021) 1–210.

[5] P. Blanchard, E.M. El Mhamdi, R. Guerraoui, J. Stainer, Machine learning with adversaries: Byzantine tolerant gradient descent, Adv. Neural Inf. Process. Syst. 30 (2017).

[6] D. Yin, Y. Chen, R. Kannan, P. Bartlett, Byzantine-robust distributed learning: Towards optimal statistical rates, in: International Conference on Machine Learning, PMLR, 2018, pp. 5650–5659.

[7] M. Fang, X. Cao, J. Jia, N. Gong, Local model poisoning attacks to {Byzantine-robust} federated learning, in: 29th USENIX Security Symposium, USENIX Security 20, 2020, pp. 1605–1622.

[8] M. Kaheni, M. Lippi, A. Gasparri, M. Franceschelli, Selective trimmed average: A resilient federated learning algorithm with deterministic guarantees on the optimality approximation, IEEE Trans. Cybern. (2024).

[9] Z. Chen, C. Yang, M. Zhu, Z. Peng, Y. Yuan, Personalized retrogress-resilient federated learning toward imbalanced medical data, IEEE Trans. Med. Imaging 41 (12) (2022) 3663–3674.

[10] N. Pang, X. Zhao, W. Zeng, J. Wang, W. Xiao, Personalized federated relation classification over heterogeneous texts, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023, pp. 973–982.

[11] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, V. Chandra, Federated learning with non-iid data, 2018, arXiv preprint arXiv:1806.00582.

[12] N. Qin, J. Du, Y. Zhang, D. Huang, B. Wu, Fault diagnosis of multi-railway high-speed train bogies by improved federated learning, IEEE Trans. Veh. Technol. (2023).

[13] J. Xiao, C. Du, Z. Duan, W. Guo, A novel server-side aggregation strategy for federated learning in non-iid situations, in: 2021 20th International Symposium on Parallel and Distributed Computing, ISPDC, IEEE, 2021, pp. 17–24.

[14] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[15] V. François-Lavet, P. Henderson, R. Islam, M.G. Bellemare, J. Pineau, et al., An introduction to deep reinforcement learning, Found. Trends® Mach. Learn. 11 (3–4) (2018) 219–354.

[16] T.D. Nguyen, P. Rieger, R. De Viti, H. Chen, B.B. Brandenburg, H. Yalame, H. Möllering, H. Fereidooni, S. Marchal, M. Miettinen, et al., FLAME: Taming backdoors in federated learning, in: 31st USENIX Security Symposium, USENIX Security 22, 2022, pp. 1415–1432.

[17] V. Tolpegin, S. Truex, M.E. Gursoy, L. Liu, Data poisoning attacks against federated learning systems, in: Computer Security–ESORICS 2020: 25th European Symposium on Research in Computer Security, ESORICS 2020, Guildford, UK, September 14–18, 2020, Proceedings, Part I 25, Springer, 2020, pp. 480–501.

[18] M. Duan, D. Liu, X. Chen, Y. Tan, J. Ren, L. Qiao, L. Liang, Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications, in: 2019 IEEE 37th International Conference on Computer Design, ICCD, IEEE, 2019, pp. 246–254.

[19] L. Wang, S. Xu, X. Wang, Q. Zhu, Addressing class imbalance in federated learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 10165–10173.

[20] X. Shuai, Y. Shen, S. Jiang, Z. Zhao, Z. Yan, G. Xing, BalanceFL: Addressing class imbalance in long-tail federated learning, in: 2022 21st ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN, IEEE, 2022, pp. 271–284.

[21] H. Wang, Z. Kaplan, D. Niu, B. Li, Optimizing federated learning on non-iid data with reinforcement learning, in: IEEE INFOCOM 2020-IEEE Conference on Computer Communications, IEEE, 2020, pp. 1698–1707.

[22] J. Shi, W. Wan, S. Hu, J. Lu, L.Y. Zhang, Challenges and approaches for mitigating Byzantine attacks in federated learning, in: 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom, IEEE, 2022, pp. 139–146.

[23] J. Peng, Z. Wu, Q. Ling, T. Chen, Byzantine-robust variance-reduced federated learning over distributed non-iid data, Inform. Sci. 616 (2022) 367–391.

[24] S. Prakash, A.S. Avestimehr, Mitigating Byzantine attacks in federated learning, 2020, arXiv preprint arXiv:2010.07541.

[25] Y. Jiang, B. Ma, X. Wang, G. Yu, C. Sun, W. Ni, R.P. Liu, A secure aggregation for federated learning on long-tailed data, 2023, arXiv preprint arXiv:2307.08324.

[26] S. Han, H. Ding, S. Zhao, S. Ren, Z. Wang, J. Lin, S. Zhou, Practical and robust federated learning with highly scalable regression training, IEEE Trans. Neural Netw. Learn. Syst. (2023).

[27] J. Zhao, H. Zhu, F. Wang, R. Lu, E. Wang, L. Li, H. Li, VFLR: An efficient and privacy-preserving vertical federated framework for logistic regression, IEEE Trans. Cloud Comput. (2023).

[28] W. Wu, Y. Zhang, An efficient intrusion detection method using federated transfer learning and support vector machine with privacy-preserving, Intell. Data Anal. (2023) 1–21, Preprint.

[29] N. Garg, K.S. Ashrith, G.S. Parveen, K.G. Sai, A. Chintamaneni, F. Hasan, Self-driving car to drive autonomously using image processing and deep learning, Int. J. Res. Eng. Sci. Manag. 5 (1) (2022) 125–132.

[30] L.M. Francis, N. Sreenath, Robust scene text recognition: Using manifold regularized twin-support vector machine, J. King Saud Univ.-Comput. Inf. Sci. 34 (3) (2022) 589–604.

[31] X. Wu, F. Huang, Z. Hu, H. Huang, Faster adaptive federated learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37, 2023, pp. 10379–10387.

[32] A. Mitra, R. Jaafar, G.J. Pappas, H. Hassani, Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients, Adv. Neural Inf. Process. Syst. 34 (2021) 14606–14619.

[33] Z. Wang, H. Xu, J. Liu, H. Huang, C. Qiao, Y. Zhao, Resource-efficient federated learning with hierarchical aggregation in edge computing, in: IEEE INFOCOM 2021-IEEE Conference on Computer Communications, IEEE, 2021, pp. 1–10.

[34] X. Li, K. Huang, W. Yang, S. Wang, Z. Zhang, On the convergence of fedavg on non-iid data, 2019, arXiv preprint arXiv:1907.02189.

[35] V. Mnih, K. Kavukcuoglu, D. Silver, A.A. Rusu, J. Veness, M.G. Bellemare, A. Graves, M. Riedmiller, A.K. Fidjeland, G. Ostrovski, et al., Human-level control through deep reinforcement learning, Nature 518 (7540) (2015) 529–533.

[36] K. Arulkumaran, M.P. Deisenroth, M. Brundage, A.A. Bharath, Deep reinforcement learning: A brief survey, IEEE Signal Process. Mag. 34 (6) (2017) 26–38.

[37] S.S. Kadam, A.C. Adamuthe, A.B. Patil, CNN model for image classification on MNIST and fashion-MNIST dataset, J. Sci. Res. 64 (2) (2020) 374–384.

[38] X. Zhang, The AlexNet, lenet-5 and VGG NET applied to CIFAR-10, in: 2021 2nd International Conference on Big Data & Artificial Intelligence & Software Engineering, ICBASE, IEEE, 2021, pp. 414–419.

[39] H.T. Nguyen, N.C. Luong, J. Zhao, C. Yuen, D. Niyato, Resource allocation in mobility-aware federated learning networks: A deep reinforcement learning approach, in: 2020 IEEE 6th World Forum on Internet of Things, WF-IoT, IEEE, 2020, pp. 1–6.

[40] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller, Playing atari with deep reinforcement learning, 2013, arXiv preprint arXiv:1312.5602.

[41] F. Colosimo, F. De Rango, Median-krum: A joint distance-statistical based Byzantine-robust algorithm in federated learning, in: Proceedings of the Int'l ACM Symposium on Mobility Management and Wireless Access, 2023, pp. 61–68.

[42] X. Hao, C. Lin, W. Dong, X. Huang, H. Xiong, Robust and secure federated learning against hybrid attacks: A generic architecture, IEEE Trans. Inf. Forensics Secur. (2023).

[43] Y. Li, A.S. Sani, D. Yuan, W. Bao, Enhancing federated learning robustness through clustering non-IID features, in: Proceedings of the Asian Conference on Computer Vision, 2022, pp. 41–55.

[44] P. Kukreja, V. Mahendran, PraaKrum: A practical Byzantine-resilient federated learning algorithm, in: 2024 16th International Conference on Communication Systems & Networks, COMSNETS, IEEE, 2024, pp. 936–944.

[45] G. Cohen, S. Afshar, J. Tapson, A. Van Schaik, EMNIST: Extending MNIST to handwritten letters, in: 2017 International Joint Conference on Neural Networks, IJCNN, IEEE, 2017, pp. 2921–2926.

[46] V. Thakkar, S. Tewary, C. Chakraborty, Batch normalization in convolutional neural networks—A comparative study with CIFAR-10 data, in: 2018 Fifth International Conference on Emerging Applications of Information Technology, EAIT, IEEE, 2018, pp. 1–5.

[47] G. Baruch, M. Baruch, Y. Goldberg, A little is enough: Circumventing defenses for distributed learning, Adv. Neural Inf. Process. Syst. 32 (2019).

[48] V. Shejwalkar, A. Houmansadr, Manipulating the Byzantine: Optimizing model poisoning attacks and defenses for federated learning, in: NDSS, 2021.

[49] X. Sun, P. Zhang, J.K. Liu, J. Yu, W. Xie, Private machine learning classification based on fully homomorphic encryption, IEEE Trans. Emerg. Top. Comput. 8 (2) (2018) 352–364.

[50] B. Knott, S. Venkataraman, A. Hannun, S. Sengupta, M. Ibrahim, L. van der Maaten, Crypten: Secure multi-party computation meets machine learning, Adv. Neural Inf. Process. Syst. 34 (2021) 4961–4973.

[51] K. Wei, J. Li, M. Ding, C. Ma, H.H. Yang, F. Farokhi, S. Jin, T.Q. Quek, H.V. Poor, Federated learning with differential privacy: Algorithms and performance analysis, IEEE Trans. Inf. Forensics Secur. 15 (2020) 3454–3469.

Yanna Jiang received her B.Sc. and M.Sc. degrees from Zhejiang University, China, in 2017 and 2020, respectively. She is pursuing a Ph.D. degree at the University of Technology Sydney, Australia. Her research interests include security and privacy in distributed learning.

Baihe Ma received his B.E. and M.E. degrees from Xidian University China in 2016 and 2019, and Ph.D. degrees from University Technology Sydney, Australia in 2024. His research interests include cybersecurity, data privacy, location privacy, and machine learning

Xu Wang (Member, IEEE) received his B.E. degree from Beijing Information Science and Technology University, Beijing, China, in 2010, and dual Ph.D. degrees from the Beijing University of Posts and Telecommunications, Beijing, in 2019, and the University of Technology Sydney, Ultimo, NSW, Australia, in 2020. He is currently a Senior Lecturer with the School of Electrical and Data Engineering, University of Technology Sydney. His research interests include cybersecurity, blockchain, privacy, and network dynamics.

Guangsheng Yu received his B.Sc. and M.Sc. degrees from the University of New South Wales Sydney, Sydney, NSW, Australia, in 2014 and 2015, respectively, and the Ph.D. degree from the Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW, Australia, in 2021. He is currently a Post-Doctoral Research Fellow with Data61, CSIRO, Sydney. His main research interests lie in blockchain and federated learning.

Caijun Sun received his B.E. degree from Hangzhou Normal University, Hangzhou, China, in 2013, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2020. He is currently a Senior Security Engineer with the Zhejiang Lab, Hangzhou. His research interests include malware analysis and data security.

Wei Ni (Fellow, IEEE) received his B.E. and Ph.D. degrees in communication science and engineering from Fudan University, Shanghai, China, in 2000 and 2005, respectively. He is a Principal Research Scientist with CSIRO and a Conjoint Professor with the University of New South Wales. He is also an Adjunct Professor with the University of Technology Sydney and an Honorary Professor with Macquarie University. He was a Post-Doctoral Fellow with Shanghai Jiaotong University, from 2005 to 2008; the Deputy Project Manager with Bell Labs, Alcatel/Alcatel-Lucent, from 2005 to 2008; and a Senior Researcher with Nokia, from 2008 to 2009. He has (co)authored one book, ten book chapters, more than 300 journal articles, more than 100 conference papers, 26 patents, and ten standard proposals accepted by IEEE. His research interests include machine learning, online learning, stochastic optimization, and their applications to the security, integrity, and efficiency of network systems. He served as the Chair for the IEEE Vehicular Technology Society New South Wales Chapter, from 2020 to 2022, the Secretary and then the Vice-Chair for the Chapter, from 2015 to 2019; the Track Chair for VTC Spring 2017; the Track Co-Chair for IEEE VTCSpring 2016; the Publication Chair for BodyNet 2015; and the Student Travel Grant Chair for WPMC 2014. He has been an Editor of *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY*, since 2022; *IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS*, since 2018; *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, since 2024; *IEEE COMMUNICATIONS SURVEYS AND TUTORIALS*, since 2024; and *Cambridge Press New Research Directions: Cyber-Physical Systems*, since 2022.

Ren Ping Liu (Senior Member, IEEE) received his B.E. degree in telecommunication engineering and the M.E. degree in computer engineering from Beijing University of Posts and Telecommunications, Beijing, China, and the Ph.D. degree in electrical and computer engineering from the University of Newcastle, Callaghan, NSW, Australia, in 1985, 1988, and 1996, respectively. He is a Professor and the Head of Discipline of Network and Cybersecurity, University of Technology Sydney, Ultimo, NSW. As a Research Leader, a Certified Network Professional, and a Full Stack Web Developer, he has delivered networking and cybersecurity solutions to government agencies and industry customers. He has supervised over 30 Ph.D. students and has over 200 research publications. His research interests include wireless networking, 5G, IoT, vehicular networks, 6G, cybersecurity, and blockchain. He was the winner of NSW iAwards 2020 for leading the BeFAQT (Blockchain enabled Fish provenance And Quality Tracking) Project. He was awarded the Australian Engineering Innovation Award 2012 and the CSIRO Chairman’s Medal for his contribution in the Wireless Backhaul Project. He was the Founding Chair of IEEE NSW VTS Chapter.