Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2023.031466

ARTICLE

Improving Federated Learning through Abnormal Client Detection and Incentive

Hongle Guo[1,2], Yingchi Mao[1,2,*], Xiaoming He[1,2], Benteng Zhang[1,2], Tianfu Pang[1,2] and Ping Ping[1,2]

[1]College of Computer and Information, Hohai University, Nanjing, 211100, China

[2]Key Laboratory of Water Big Data Technology of Ministry of Water Resources, Hohai University, Nanjing, 211100, China

*Corresponding Author: Yingchi Mao. Email: yingchimao@hhu.edu.cn

Received: 19 June 2023 Accepted: 28 September 2023 Published: 30 December 2023

ABSTRACT

Data sharing and privacy protection are made possible by federated learning, which allows for continuous model parameter sharing between several clients and a central server. Multiple reliable and high-quality clients must participate in practical applications for the federated learning global model to be accurate, but because the clients are independent, the central server cannot fully control their behavior. The central server has no way of knowing the correctness of the model parameters provided by each client in this round, so clients may purposefully or unwittingly submit anomalous data, leading to abnormal behavior, such as becoming malicious attackers or defective clients. To reduce their negative consequences, it is crucial to quickly detect these abnormalities and incentivize them. In this paper, we propose a Federated Learning framework for Detecting and Incentivizing Abnormal Clients (FL-DIAC) to accomplish efficient and security federated learning. We build a detector that introduces an auto-encoder for anomaly detection and use it to perform anomaly identification and prevent the involvement of abnormal clients, in particular for the anomaly client detection problem. Among them, before the model parameters are input to the detector, we propose a Fourier transform-based anomaly data detection method for dimensionality reduction in order to reduce the computational complexity. Additionally, we create a credit score-based incentive structure to encourage clients to participate in training in order to make clients actively participate. Three training models (CNN, MLP, and ResNet-18) and three datasets (MNIST, Fashion MNIST, and CIFAR-10) have been used in experiments. According to theoretical analysis and experimental findings, the FL-DIAC is superior to other federated learning schemes of the same type in terms of effectiveness.

KEYWORDS

Federated learning; abnormal clients; incentive; credit score; abnormal score; detection

1 Introduction

With Internet of Things (IoT) technology continuing to evolve, a huge amount of data is being generated at the edge and Machine Learning (ML) is being used as a powerful analytical tool in Internet of Things scenarios [1]. Traditional machine learning frameworks collect data from different information sources, and then the data are shared with the central server for processing, after which the machine learning algorithms are trained in the central server. However, this centralized framework has

two drawbacks. Firstly, data is shared with the central server, and client privacy may be compromised. Secondly, due to the huge amount of shared data, the communication and time overhead become expensive.

In this regard, the development of federated learning both safeguards client privacy and minimizes information transmitted [2]. Instead of sharing the original data, the federated learning model is trained locally, and then the model parameters are shared with the central server. As a result, the client's original data is left locally, and much less information is transmitted over the network. The Federated Averaging algorithm (FedAvg) is one of the most widely used federated learning algorithms at the moment [3].

The FedAvg framework was proposed by McMahan et al. This framework allows the averaging of different client aggregations. However, in practice, it is difficult to ensure that all clients are normal throughout the training process, considering that federated learning usually requires thousands of rounds of communication to converge and the number of clients involved in training is relatively large. Additionally, unlike distributed machine learning, federated learning does not allow the central server to view data provided by clients or control client behavior. Therefore, clients may experience abnormal behavior in federated learning which is initially referred to as a Byzantine attack. In this paper, we call this abnormal client behavior, as shown in Fig. 1. Abnormal client behavior may be caused by a malicious attacker's attack, or it may be due to the client's own will. It is now especially crucial to reduce the impact of abnormal clients.

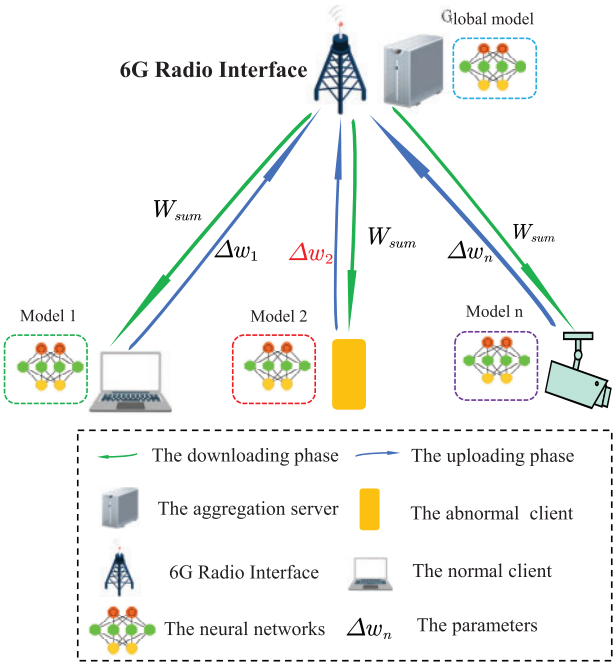


Figure 1: Federated learning framework in the Internet of Things, where the yellow squares represent anomaly clients

However, the traditional Byzantine fault-tolerant algorithms are based on defense, they employ untargeted defense against attackers at the expense of honest clients, and this approach tends to reduce model accuracy. Therefore, there is a lack of effective anomalous client detection schemes to prevent anomalous clients from reducing the model training rate. In addition, if abnormal clients are detected,

previous work has been to discard them directly, but discarding too many clients leads to missing information as well as unfairness problems, which results in low global model prediction performance. The client's identity as the normal client and the abnormal client are interchangeable. Such clients are active during round t but may become abnormal clients at $t+1$. Or they are abnormal clients in round t but become normal clients again in round $t+1$. Then, effective incentives are needed to leave more clients to provide quality data.

Therefore, to efficiently execute federated learning in response to the presence of problems with abnormal clients, we propose a Federated Learning scheme for Detecting and Incentivizing Abnormal Clients (FL-DIAC). Specifically, we propose a Detecting Anomalous Clients Module Based on Detectors (DACM-D). This module uses a pre-trained anomaly detection model to detect anomalous client behavior and eliminate its adverse effects. We apply the FedAvg algorithm to aggregate model weight updates. Since the model weights of deep learning models can easily be too large, we use a downscaling technique to generate a proxy for local model weight updates on the server for anomaly detection. In addition, to encourage more clients to participate in federated learning, we propose an incentive mechanism called Client Incentive Module Based on Credit Score (CIM-CS) to encourage clients to participate in training.

The following are the main contributions of this paper:

- To achieve the efficiency and security of federated learning, we propose a scheme to detect anomaly clients and design incentive mechanisms to improve federated learning. Specifically, an approach for detecting abnormal clients based on the detector that uses an anomaly detection model to detect abnormal client behavior and eliminate its negative effects is proposed. Additionally, to reduce the computational complexity, a Fourier transform-based anomaly data detection method for latitude reduction is proposed.
- In addition, an incentive method based on credit score is proposed, which introduces a multi-dimensional reverse auction to encourage more clients to participate in the training.
- Three data sets (MNIST, Fashion MNIST, and CIFAR-10) and three training models (CNN, MLP, and ResNet-18) were set in our experiments. The experimental results demonstrate that the FL-DIAC scheme can increase the client participation rate and ensure the accuracy of model training compared to the same type of federated learning schemes.

The rest of this paper is organized as follows. We review related work in Section 2. In Section 3.1, we introduce the overall framework of FL-DIAC. We present the DACM-D model of FL-DIAC in detail in Section 3.2 and the CIM-CS model of FL-DIAC in detail in Section 3.3. In Section 4, we conduct extensive experiments to evaluate the performance of FL-DIAC. Finally, we conclude the paper and guide our future work in Section 5.

2 Related Work

In this section, we review the related works on detecting anomalous clients and client incentives in order to relate our study to existing research.

2.1 Detecting Abnormal Clients

The defense of anomalous clients in the context of federated learning has received a lot of attention. The majority of works are defensive in nature. GeoMed [4], Krum [5], and Trimmed Mean [6] are a few examples. A gradient update method for distribution learning of heterogeneous distributed data that is resistant to Byzantine attacks is also introduced by Li et al. [7] called RSA. Under federated

learning, a distributed backdoor defense technique is proposed by Shen et al. [8]. In federated learning, distributed backdoor assaults are a major issue that is mostly addressed by this study. However, the existing approaches employ untargeted defense against attackers at the expense of honest clients, and this approach tends to reduce model accuracy.

2.2 Incentive Abnormal Clients

To incentivize more clients to take part in training for federated learning, several works of literature have designed incentive mechanisms from different perspectives to increase client engagement. Ding et al. [9] chosed a dimension reduction technology to consider the optimal pricing scheme based on the variability of the client information possessed by the server, and this scheme solves the information asymmetry problem between the client and the server. For clients with different data privacy preserving needs, Wu et al. [10] classified the contribution and privacy overheads of the client and established the payment mechanism corresponding to the privacy type through contract theory. Sarikaya et al. [11] analyzed the reasons for the heterogeneous impact of federated learning clients and proposed an incentive mechanism based on a master-slave type to balance the time delay of each iteration. Ding et al. [12] proposed a multidimensional contract scheme to build the best optimal incentive mechanism on the server side to obtain the best data volume and shortest communication time by solving the optimal reward for various data and communication volumes. This scheme takes into account different multidimensional privacy information of clients including training overheads, transmission delay, and data volume. In order to allow model transmission and free trading, Feng et al. [13] built a cooperative communication platform based on relay networks, where clients are rewarded for acting as relay nodes, and models are passed to the server over a cooperative relay network.

The auction mechanism is also applied to federated learning. However, this may degrade the performance of federated learning due to heterogeneity between different clients. A brand-new multidimensional incentive paradigm for federated learning was put forth by Zeng et al. [14]. The best policy for each client is determined using game theory, and the server and client are jointly parameterized to choose the best client to train the model.

Another useful tool for assessing data contributions is contract theory. In a scenario involving multiple clients, Lim et al. [15] proposed a tiered incentive architecture. They create incentives between clients and users using contract theory, and they are simply rewarded for the actual contribution of their marginal clients.

Additionally, Zhan et al. [16] suggested a theory based on an incentive mechanism for a federated learning scheme that integrates distributed machine learning with swarm intelligence perception for large data analysis on mobile clients. This incentive mechanism adjusts the quantity of the data used in training. The platform begins by distributing a task and its associated reward. Each edge client determines its degree of engagement, or the volume of training data, to maximize its utility, taking into account the rewards it receives and the energy expenses. The edge client's decision-making dilemma is treated as a non-cooperative game in order to establish a Nash equilibrium.

Deep Reinforcement Learning (DRL) is another effective strategy. Zhan et al. [17,18] suggested an incentive mechanism for DRL-based federated learning by combining deep reinforcement learning and game theory. This study characterizes the shared between the central server and the clients as a Starkelberg game to encourage clients to take part in model training. The client chooses the amount of data to participate in the training as a follower when the central server publishes a training assignment and announces a total reward for the task leader.

Zhang et al. [19] used the Starkelberg game and auction theory, respectively, to study central server-centric and client-centric crowdsourcing. To encourage users to employ inter-device communication, Li et al. [20] suggested incentive schemes. They took into account two alternative informational settings, one where clients have access to information about all other clients and the other where clients only have access to their own information. Zhan et al. [21] created online and offline methodologies as well as incentive mechanisms for opportunity networks.

While the above initiatives work to focus on incentive-related concerns, they only take into account one aspect of pricing and contribution.

In summary, traditional Byzantine tolerance algorithms are defense-based and the performance of such schemes is inefficient. The crux of the performance degradation is that the existing methods employ untargeted defenses to defend against attackers at the expense of honest clients. Abnormal clients continue to exist. Therefore, the main objective of this paper is to detect abnormal clients and eliminate the impact of abnormal clients on accuracy. In addition, we can precisely build a model of each participant's contribution when designing incentives in these areas. Furthermore, when designing the federated learning incentive mechanism, it is important to estimate the value of training data for each client, and cut costs. So, using the aforementioned two factors pricing and contribution, we will develop the incentive mechanism in this study.

3 Scheme Architecture

We introduce each FL-DIAC component in this section.

3.1 Scheme Model

The FL-DIAC architecture comprises two physical bodies: the client and the central server, as depicted in Fig. 2. The central server manages the gathering and aggregating parameters that the client uploads and the client is the IoT's data owner. Moreover, FL-DIAC is made up of two components. Specifically, while aggregating models, the central server is unable to identify the anomalous clients since standard federated learning cannot detect the abnormalities of model parameters submitted by clients. We propose Detecting Anomalous Clients Module Based on Detectors (DACM-D) to achieve the objective that anomalous clients can be detected, maximize the performance of the global model, and reflect its credit score for each client to support the subsequent incentive for clients to participate in training. Additionally, we propose a credit score-based client incentive mechanism called Client Incentive Module Based on Credit Score (CIM-CS) to encourage abnormal clients to actively take part in the next epoch of training. For ease of reading, the article's most significant notations are included in Table 1 below.

The complete illustration of the FL-DIAC framework is shown in Fig. 3. Step 1: some clients download the global model parameters from the central server; Step 2: the parameters uploaded by the local clients are detected by the Detecting Anomalous Clients Module based on Detector, and the normal clients are selected; Step 3: the abnormal clients are incentivized by the Client Incentive Module based on Credit Score, which makes them actively participate in the training; Step 4: the central server performs the aggregation update. The above iterations are repeated for each round until the set number of iteration rounds or the desired model accuracy is reached.

Below is a description of the particulars of each of these two modules.

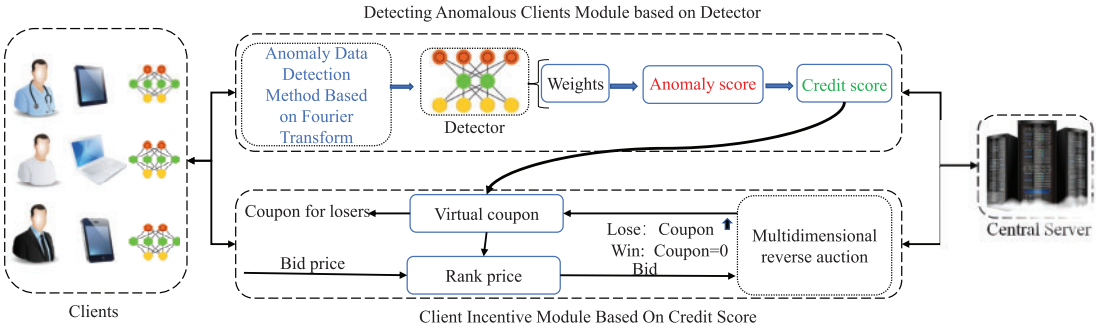


Figure 2: Architecture of the FL-DIAC

Table 1: Notation

Notation	Description
P	The aggregate server
C_i	The clients
W	The set of parameters
t	The training round
w_i	The model parameter of client i in the W
w_0	The initialized model parameters
η	The learning rate
D	The client dataset
$epoch$	The upper limit of training rounds
d_i	The virtual coupon
b_i	The actual bid price
p_i	The ranked price

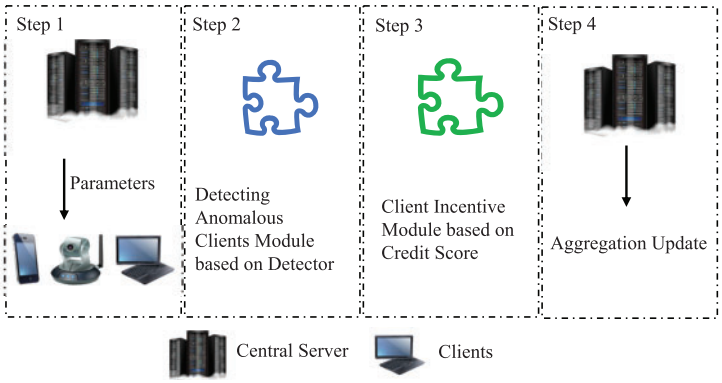


Figure 3: The complete illustration of the FL-DIAC

3.2 Detecting Anomalous Clients Module Based on Detectors

This article makes the following assumptions about the traditional federated learning framework. Specifically, in this scenario, the central server is fully trusted, but anomalies can occur on the client side. In addition, the standard verification set on the central server side is selected according to the needs of the task model owner. This part of the data is not accessible to the client. The encryption and decryption process of model transmission is omitted in the experiments because the encryption and decryption are not relevant to the scenario proposed in this paper.

Algorithm 1: Detector-based algorithm for abnormal client detection

Input: $w_i\ (i = 1, 2, 3, ..., n)$, *epoch*

Output: w_{sum}

1: **for** $i \leq$ epoch **do**

2: Clients: $w_i \leftarrow w_i - \eta\ (w{:}b)$

3: Fourier transform based anomaly detection method for dimensionality reduction: $\left\{w_i^{epoch}\right\}\ (i = 1, 2, 3, ..., n) \overset{FFT}{\leftarrow} (w_i\ (i = 1, 2, 3, ..., n))$

4: Reconstruction error calculated: $Err\left(w_{t+1}^i\right) = \|w_{t+1}^i - \tilde{w}_{t+1}^i\|^2$

5: Abnormal score calculated: $A_{t+1}^k = \dfrac{1 + Err\left(w_{t+1}^k\right)}{1 + \sigma_{t+1}}$

6: Credit score calculated: $\alpha_{t+1}^k = \dfrac{n_k\left(A_{t+1}^k\right)^{-L}}{\sum_{j=1}^K n_j\left(A_{t+1}^j\right)^{-L}}, \forall j = 1, 2, \cdots, K$

7: Get: $w_{sum} \leftarrow w_{t+1} = \sum_{k=1}^K \alpha_{t+1}^k w_{t+1}^k$.

8: **end for**

In Fig. 4, DACM-D has two entities: N clients and a central server. Each client receives the model parameters sent from the central server and then trains the model using its local dataset and uploads the parameters of the new training model to the server. Unlike traditional federated learning, we design a detector in the server to calculate a credit score for each model parameter uploaded by the client. The detector is a deep artificial neural network. The credit score of each client is calculated by the detector. Specifically, as shown in DACM-D, the server collects the local model parameters uploaded by each client and then uses the model parameters as the input to the detector, whose output is the credit score of each local model. In addition, before the model parameters are input to the detector, we propose a Fourier transform-based anomaly data detection method for dimensionality reduction to reduce the computational complexity. The central server accepts all model parameters and selects those with high credit scores for aggregation according to the federated average method to update the global model. Algorithm 1 displays the Detector-based algorithm for abnormal client detection. The following is a more thorough description.

Initialization The central server establishes the federated learning architecture and initializes the global model's parameters at random. The initialized model parameters w_0 are then transmitted to each client by the central server.

In upload phase In the training round t, each client is trained using its local data combined with the downloaded newest shared parameters w_i^t, and updated to obtain the newest local parameters w_i^t. Then, each client uploads the newest local model parameters w_i^{t+1} to the central server. At this point, the minimize the empirical loss $F\left(w_i^t\right)$ is calculated about client i in epoch t.

$$w_i^t = \arg\min_{w_i^t} F(w_i^t), \tag{1}$$

$$F\left(w_i^t\right) = \frac{1}{|D_i|}\sum_{j\in D_i} f_j(w_i^t), \tag{2}$$

where D_i denotes the client's data set. $|D_i|$ denotes the amount of samples.

The client-side update method uses Stochastic Gradient Descent (SGD) [22].

$$w_i^{t+1} = w_i^t - \eta\nabla F\left(w_i^t\right), \tag{3}$$

where $\delta\nabla F\left(w_i^t\right)$ denotes the gradient of the loss function and η is the learning rate.

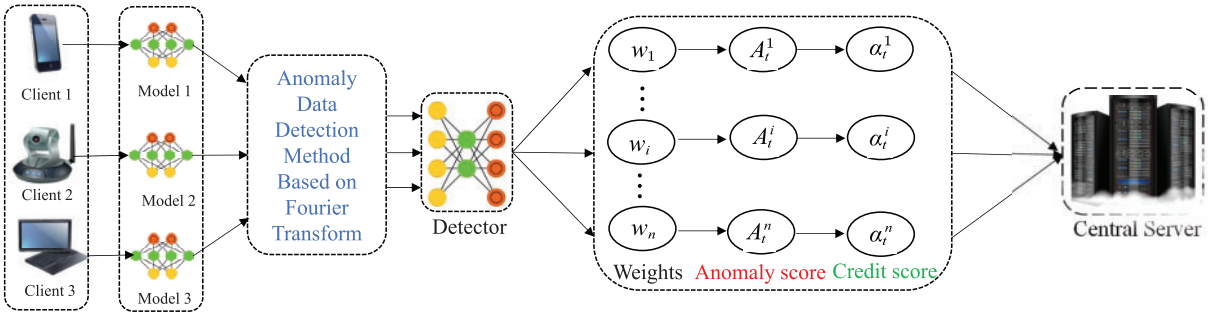


Figure 4: Detecting anomalous clients module based on detectors (DACM-D)

Detection The central server collects the model parameters shared by each client. Those parameters are introduced to the detector to determine each local model's credit score. We can dynamically select the clients involved in federated learning based on the credit scores of the model parameters to maximize the accuracy of federated learning.

Specifically, in DACM-D, the shared newest local model parameters by the client are used as input to the central server detector. On the central server, we use an already-trained autoencoder model to detect abnormal parameters from the client. Autoencoder can reconstruct and restore normal data, but it cannot restore anomalous data as well. So it is used for anomaly detection [23]. And autoencoder is better at processing high-dimensional data [24].

Our detection-based approach's main tenet is to first determine each client's anomaly score using an anomaly detection model, and then determine each client's credit score using the anomaly score, where the number of clients takes part in federated learning training is K, the amount of data each client is n_k, the local model parameter of each client is w_{t+1}^k, and the epoch is $t+1$.

The following equation provides the formula for the model parameters in the FedAvg framework:

$$w_{t+1} = \sum_{k=1}^{K}\frac{n_k}{n}w_{t+1}^k, \tag{4}$$

where n_k denotes the number of clients currently involved in training, and n denotes the amount of data for all, w_{t+1} denotes the global model weight update, and we have $\sum_{k=1}^{K} n_k = n$.

In this case, the more data from the client involved in the training, the greater the weighting. Therefore, in Eq. (4), we use α_{t+1}^k instead of $\frac{n_k}{n}$ to obtain the following equation:

$$w_{t+1} = \sum_{k=1}^{K}\alpha_{t+1}^k w_{t+1}^k. \tag{5}$$

The client k uses the abnormal score A_{t+1}^{k} in epoch $t+1$ to get the credit score α_{t+1}^{k} calculated as follows:

$$\alpha_{t+1}^{k}=\frac{n_k\left(A_{t+1}^{k}\right)^{-L}}{\sum_{j=1}^{K} n_j\left(A_{t+1}^{j}\right)^{-L}}, \forall j=1,2,\cdots,K, \tag{6}$$

For the detection of abnormal clients, our scheme considers both the abnormal score and the data score. The hyperparameter L in Eq. (6) controls how much A_{t+1}^{k} influences the calculation of α_{t+1}^{k}. L is a parameter of >1. If the data volume is very unevenly distributed, we can increase the value of L. If one client owns a sizable chunk of the data, L should have a high value. In our scheme, we propose an anomaly detection model based on autoencoder [25] to obtain anomaly scores.

The training data for the autoencoder is $D=\left\{w_{-1}^{1}, w_{-1}^{2}, \cdots, w_{-1}^{N}\right\}$, which represents the parameters after the last epoch of central model aggregation. Therefore, the subscript is -1. Using this dataset D, the autoencoder is already trained. The input of the autoencoder is the model parameters uploaded by each client w_{t+1}^{i}. The output of the autoencoder is $\tilde{w}_{t+1}^{i}$ [26]. Given by is the reconstruction error of the i.

$$Err\left(w_{t+1}^{i}\right)=\|w_{t+1}^{i}-\tilde{w}_{t+1}^{i}\|^{2}. \tag{7}$$

Afterward, we can get the anomaly score. In the epoch $t+1$, client k received A_{t+1}^{k} as

$$A_{t+1}^{k}=\frac{1+Err\left(w_{t+1}^{k}\right)}{1+\sigma_{t+1}}, \tag{8}$$

where $\sigma_{t+1}=\min_j\left\{Err\left(w_{t+1}^{j}\right), j=1,2,\cdots,K\right\}$.

In the federated learning framework, the clients participating in the training are often in the tens of thousands. As a result, the latitude of the parameters input to the autoencoder is substantial, which can lead to high computational complexity for anomaly client detection [27]. To reduce the computational complexity, Ghosh et al. [28] used a random element extraction method for dimensionality reduction, and this dimensionality reduction method may leave some normal model parameters of the client unextracted, thus causing a decrease in the quasi-deficiency rate. Therefore, we propose a Fourier transform-based anomaly data detection method. We first detect the input model parameters and kick out the clients that are particularly abnormal, thus generating low-dimensional vectors that are then used as input to the detector [29].

The Fourier transform based on the anomaly data detection method is described as follows.

The Fourier transform converts digital signals in the time domain into frequency domain signals. We consider each round of shared model parameters as a signal with n clients corresponding to n sampled values. We put the n clients through the Fourier transform to observe the differences between each of these model parameters, find the anomalies in them, and determine the anomalous model parameters.

The Fast Fourier Transform is a commonly used method, and we use the Fast Fourier Transform to complete the transformation. First, we take the n model parameters according to the requirements of Fast Fourier Transform, the number must meet 2^m, take the closest 64, and if the number does not reach 64 is taken as 0, this process is called "data signal expression". Second, for each household to carry out Fast Fourier Transform transformation, this process directly uses the Fast Fourier Transform algorithm, without any improvement to the finally, the results are counted to find the abnormal model for kicking out.

After obtaining the client's abnormal score, the credit score of each client is calculated by using the Eq. (6). Then we use the threshold value to select the anomalous clients, and this process is the same as the two-mechanism classification process. Here, we will set a threshold value A_{t+1}^{th}. This threshold can be either the median or the mean. For any client whose model parameter has an anomaly score A_{t+1}^{k} greater than the threshold value for this epoch, this model parameter is not selected for aggregation in this epoch. Therefore, the client providing this model parameter is temporarily defined as an anomaly client in this epoch. The credit score α_{t+1}^{k} of the client is set to zero.

Aggregation The parameter server collects the client's local model parameters for each epoch and aggregates them by FedAvg to get the latest global model. Higher aggregation weights will be applied to models that were trained using high-value data. The global model parameters that have been combined are then forwarded to each client. The following updates are made to the newest model parameters:

$$w_i \leftarrow w_i - \eta(w_i{:}b). \tag{9}$$

Finally, each client downloads the latest model parameters from the central server for each epoch and proceeds to the next epoch until the model converges to the optimal accuracy rate.

3.3 Client Incentive Module Based on Credit Score

In the previous subsection, our goal is to detect anomalous clients and obtain a credit score for each client. Due to the presence of anomalous clients, clients suffer from privacy breaches of local data, and such clients do not have sufficient rewards for clients who may not want to participate or share their models. For the abnormal clients. The previous solution was to remove the abnormal client, but dropping the client leads to missing information as well as an unfairness problem, which leads to low performance of global model prediction. The client's identity is switched between normal and abnormal clients. Such clients are active in round t but have the possibility to become abnormal clients at t + 1. Or they are abnormal clients in round t but become normal clients again in round t + 1. Therefore, to incentivize abnormal clients to actively take part in federated learning, we propose a credit score-based client incentive mechanism and encourage them to stay passionate in training through rewards. More clients are attracted to providing high-quality models.

The proposed credit score-based incentive method for federated learning clients is shown in Fig. 5. In the CIM-CS, a multidimensional reverse auction [30] is introduced in the incentive module in order to incentivize clients. The goal of CIM-CS is to prevent abnormal clients from being discarded or dropping out of training early while reducing the incentive overheads by preventing overhead explosion during the multidimensional reverse auction.

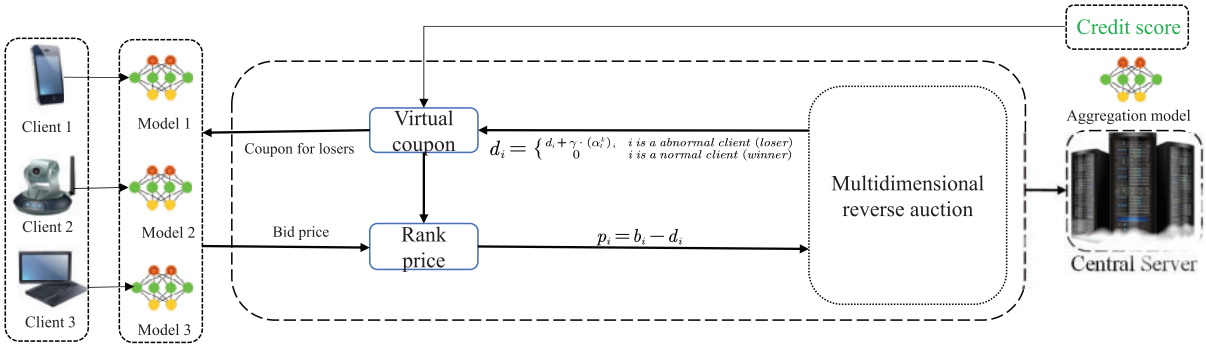


Figure 5: Client incentive module based on credit score (CIM-CS)

As shown in Fig. 6, in a one-dimensional reverse auction, because only one attribute determines the reward, it is possible that some clients that provide abnormal parameters will be the eventual winners, and these clients always are judged as the winners. Then, the client that always loses can possibly drop out of federated learning training. In addition, some clients are always judged to be losers because they are not positive at one time.

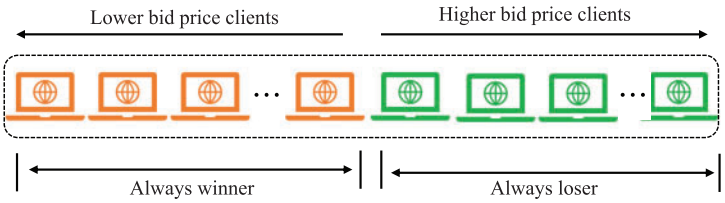


Figure 6: One-dimensional reverse auction

To make competition fair and to prevent excessive overheads of auctions, a sufficient number of clients should continuously participate in the reverse auction of CIM-CS. To this end, the proposed CIM-CS incentive module combines virtual coupons and credit scores calculated by the DACM-D module to get a novel federated learning incentive mechanism.

To boost the likelihood of winning the following auction round, client i will be given a virtual coupon as compensation for losing the previous round. Virtual coupons are expressed as $D = \{d_1, d_2, \cdots, d_n\}$ and has the following formula:

$$d_i = \begin{cases} d_i + \gamma \cdot \left(\alpha_t^k\right)_i & i \text{ is a abnormal client } (loser) \\ 0 & i \text{ is a normal client } (winner) \end{cases}, \tag{10}$$

where γ is the number of virtual coupons and $\left(\alpha_t^k\right)$ is the credit score provided by the FLDAC-D module. Thus, whenever client i fails in the auction round, the number of coupons γ weighted by the credit score $\left(\alpha_t^k\right)$ is added to the virtual coupon d_i. More virtual coupons are available to creditworthy clients.

The virtual coupon d_i is set to zero whenever the client i wins or withdraws from the previous epoch of auctions. Virtual coupons increase the probability of the client winning this epoch being selected.

We distinguish between two bid prices: the actual bid price and the ranking price. Client i proposes the actual bid price b_i. The equation can be used to calculate the ranked price p_i.

$$p_i = b_i - d_i \tag{11}$$

In the proposed incentive module, the ranking price p_i is used to select the winner in each round of the auction, and CIM-CS increases the bidder's probability of winning by using virtual coupons to reduce the ranking price.

Even participants with higher bids can become winners through continuous participation (as shown in Fig. 7). Therefore, CIM-CS encourages normal clients to continuously participate in training and abnormal clients to actively participate in training.

3.4 Complexity Analysis

For the DACM-D algorithm, the client upload parameters need to be traversed and its complexity is $O(n)$. For m dimensional data, n denotes the data length. Then the complexity of the fast Fourier transform is $O\left(m \cdot n \cdot \log^n\right)$. The complexity of the computation of the abnormal score is $O(n \log(n))$.

The complexity of the client credit score evaluation phase is $O(n \log(n))$. The complexity of the CIM-CS algorithm is also $O(n \log(n))$.

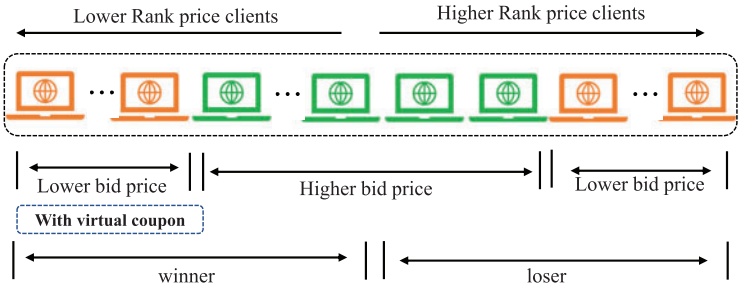


Figure 7: Combining virtual coupons with multi-dimensional reverse auctions

4 Experimental Evaluation

In this section, we use simulation settings to evaluate FL-DIAC's performance. In-depth descriptions are provided of the datasets, models, benchmarks, and experimental setup used in the experiments. The analysis and outcomes of the experiment are given.

4.1 Dataset and Model

In this section, we evaluate the performance of FL-DIAC using three widely used learning models, including Convolutional Neural Network (CNN) [31], Multi-Layer Perceptron (MLP) [32], and ResNet18 [33]. Because CNN and Multi-Layer Perceptron (MLP) have good image feature extraction capabilities, CNN and MLP are used to identify the MNIST images and the Fashion MNIST face database. The softmax cross-entropy as the loss function is used. The above three models were trained using the following datasets: as shown in Table 2, MNIST [34], Fashion MNIST [35], and CIFAR-10 [36] datasets, respectively. Because these above datasets reflect the characteristics of IoT device data. We choose the accuracy of the model on the test set as an indicator to demonstrate the effectiveness of FL-DIAC.

Table 2: Size of MNIST, fashion MNIST, and CIFAR-10 datasets

	Training	Validation	Test
MNIST	240,000	8,000	32,000
Fashion MNIST	60,000	2,000	8,000
CIFAR-10	60,000	2,000	8,000

MNIST is a significant collection of handwritten numbers that the National Institute of Standards and Technology has gathered. 10,000 photos and labels make up the test set, while 60,000 images and labels make up the training set.

Fashion MNIST is meant to be a more difficult replacement for the first MNSIT dataset. A balanced selection of 10 distinct classes, each with 7,000 samples, is present. With 60,000 training samples and 10,000 test samples, the dataset has 70,000 samples altogether.

CIFAR-10 is a subset of the 80 million micro picture collection that has been tagged. There are 60,000 32 × 32 color photos total, divided into 10 categories with 6,000 images each. There are 1,000 photographs in each category, with 50,000 training images and 10,000 test images.

4.2 Benchmark

To validate the performance of our proposed FL-DIAC, the following reasonable benchmarks were used.

FedAvg [3] is a popular model aggregation algorithm for federated learning that is effective and commonly used in federated learning frameworks today. The quantity of data samples utilized for training determines the aggregate weights provided to model updates in this algorithm.

Median [37] is a Byzantine robust federated learning, which is based on mean aggregation at the central server.

Krum [5]. The closest majority is chosen as the aggregated model after the customers are sorted according to the geometric distance of the customer model update distribution.

Bulyan [38] aggregates the leftover clients after sorting the clients by geometric distance.

4.3 Experimental Settings

The simulation experiment is run in Python 1.4.0 using Pytorch to mimic the FL-DIAC on a server with an Intel(R) Core(TM) i7-10875H CPU running at 2.30 GHz and 32 GB of RAM.

The training data is dispersed across the clients using a non-IID distribution in order to adapt the MNIST, Fashion MNIST, and CIFAR-10 datasets to the federated environment. The Dirichlet distribution is used to generate cross-category federated datasets for different clients, where the parameter $\alpha \in [0, +\infty)$ reflects the degree of heterogeneity of the generated federated datasets, where a larger α indicates a more consistent distribution of data between clients, and $\alpha = 0$ indicates that only one category of data is assigned to a single client. In this paper, a dataset with heterogeneity of $\alpha = 10$ is selected for the experiment. Therefore, to simulate a situation where each client only has incomplete knowledge, we at random assign instances with fewer labels to each client. Additionally, the presence of anomalous clients, a quick labeling of training data, and other factors could result in the training samples of federated learning clients having corrupted samples. To increase the federated task model's accuracy, it is very helpful to automatically identify datasets containing corrupted samples. In this simulation experiment, to simulate the presence of abnormal clients, we added different amounts of noise to the model parameters shared by different clients. We take into account the sign-flipping, additive noise, and gradient ascent of three adversarial assault models. A sign-flipping assault changes the sign of the model parameters. The additive noise attack increases the model parameters with Gaussian noise. The abnormal clients launch a local gradient climb attack rather than a gradient descent attack.

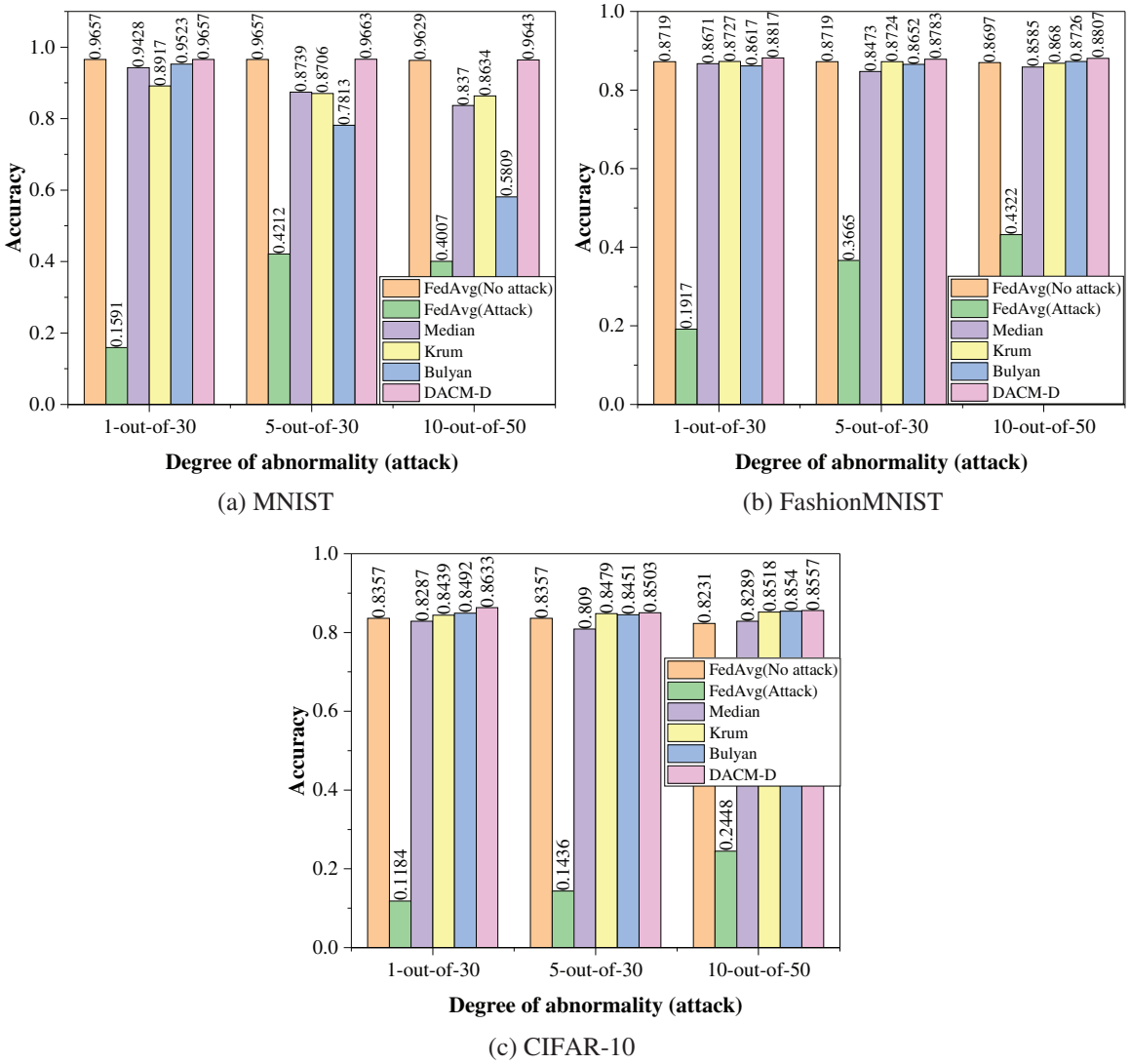
4.4 Experimental Analysis

4.4.1 Performance Analysis of DACM-D

In order to validate the performance of the DACM-D, the DACM-D focuses on detecting abnormal clients and reducing the training time. Therefore, we will compare it with the same type of methods in terms of accuracy and training time.

As shown in Figs. 8a–8c, 9a–9c, 10a–10c, the proposed module (DACM-D) outperforms the baseline schemes: Median scheme, Krum scheme, and Bulyan scheme in terms of model accuracy

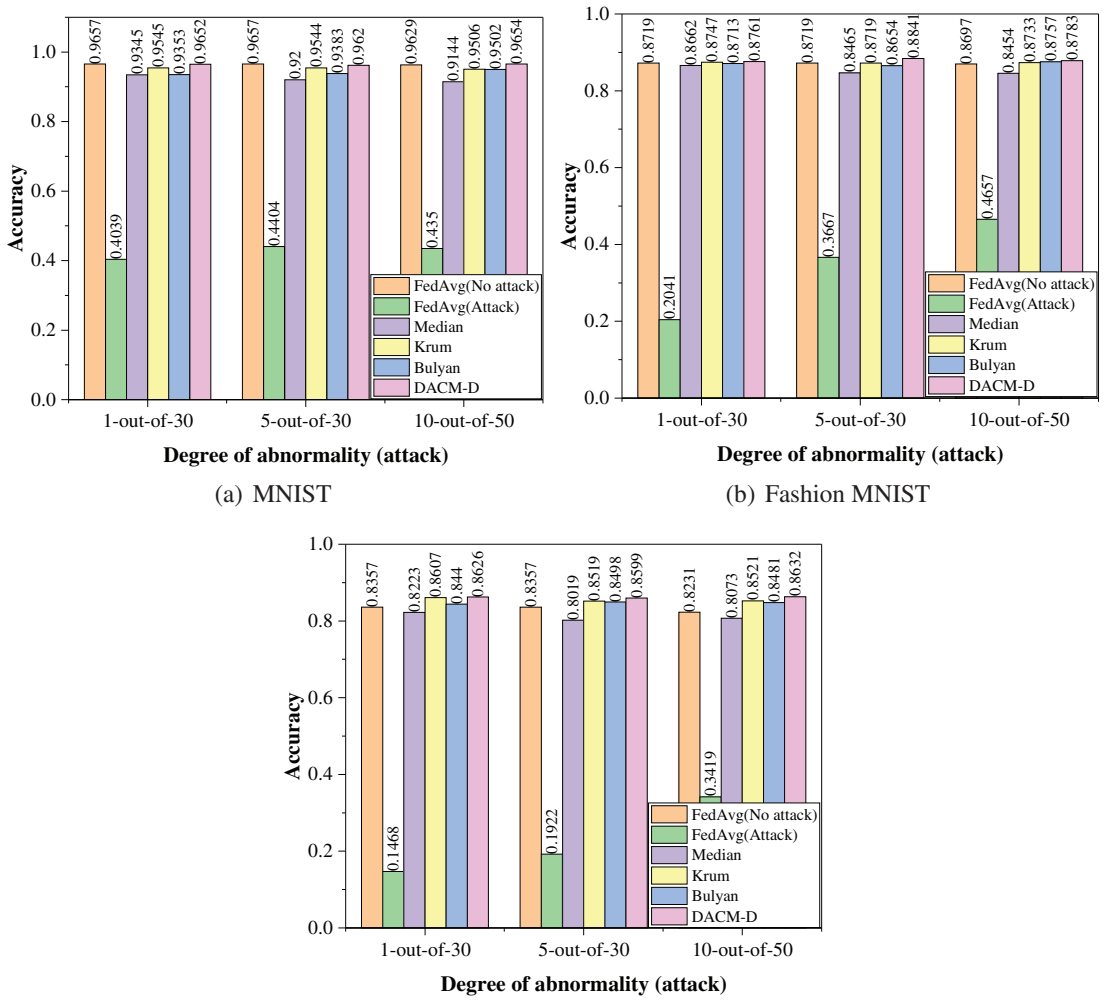
under the attacks of sign-flipping, additive noise, and gradient ascent. In the absence of abnormal clients, our proposed detector-based model DACM-D has approximately the same accuracy as the FedAvg algorithm. In the absence of any detection or defense, the FedAvg algorithm has the lowest model accuracy, which is even lower than 0.1. Although Median is better at resolving unusual client issues, the proposed detector-based model DACM-D improves the model accuracy under symbol flipping attacks by 0.01 or more than Median. Moreover, as described in [37], Median cannot defend against a large number of anomalous clients. The Krum algorithm is a federated learning scheme based on the Euclidean distance, and it is an algorithm that can still converge in the presence of abnormal clients. As a result, Krum was unable to resolve the problems that existed with the abnormal client.



(a) MNIST (b) FashionMNIST

(c) CIFAR-10

Figure 8: Model performance under sign-flipping ((a): MNIST, (b): Fashion MNIST, (c): CIFAR-10)



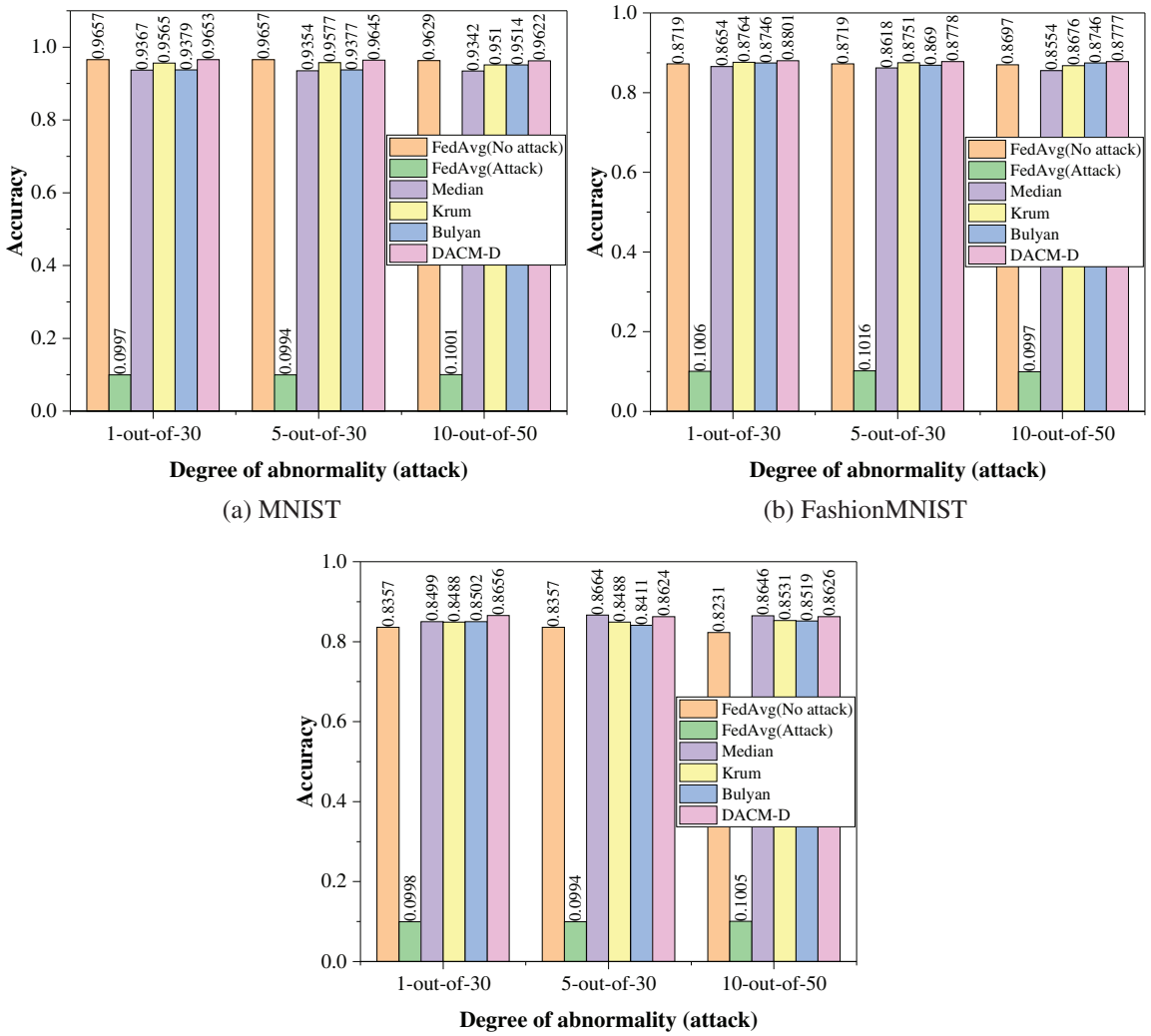
(a) MNIST (b) Fashion MNIST

(c) CIFAR-10

Figure 9: Model performance under additive noise ((a): MNIST, (b): Fashion MNIST, (c): CIFAR-10)

As shown in Table 3, the overall distribution of the training time for the four schemes remains constant under the attacks of sign-flipping, additive noise, and gradient ascent attacks. The proposed module DACM-D uses the least training time in this paper, which is mainly due to the fact that DACM-D takes into account a variety of computationally efficient clients and the low computational complexity of the detection method. The median scheme, and Krum scheme all showed an increase in training time compared to the FedAvg scheme. The above experimental results also show that DACM-D is more suitable for resource-constrained mobile edge environments.

As shown in Fig. 11a, the accuracy of the model with all clients selected for training is the lowest. The model accuracy of the Fourier transform-based abnormal data detection method is higher than that of the method without the use of Fourier transform. Similarly, it is derived from Fig. 11b that the training time by the Fourier transform-based anomaly data detection method is also minimal. Therefore, the Fourier transform-based abnormal data detection method proposed in this paper is effective.



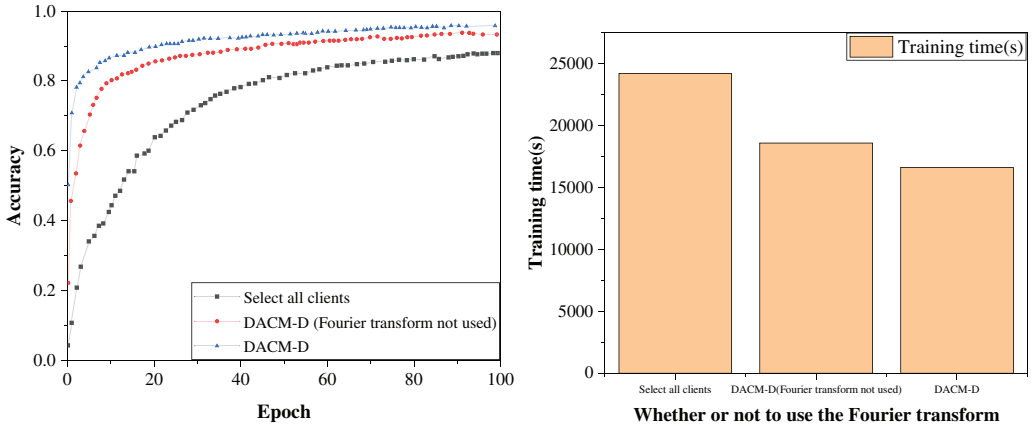
(a) MNIST (b) FashionMNIST

(c) CIFAR-10

Figure 10: Model performance under gradient ascent ((a): MNIST, (b): Fashion MNIST, (c): CIFAR-10)

Table 3: Training time (s) comparison with different schemes

Training time (s)			
Scheme	Sign-flipping	Additive noise	Gradient ascent
FedAvg	18568	18406	18523
Median	22954	22145	22682
Krum	27412	27110	27313
DACM-D	16233	16596	16831



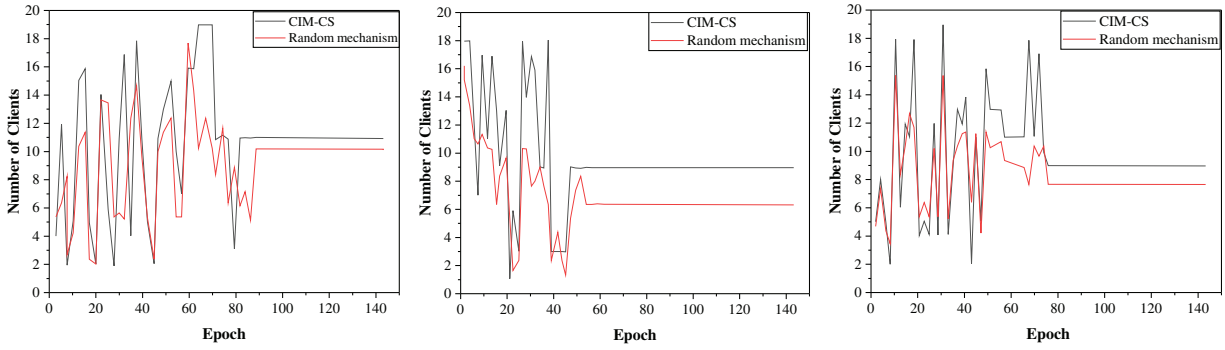
(a) Accuracy (b) Trainingtime

Figure 11: Comparison of performance of modules with and without Fourier transforms ((a): Accuracy, (b): Training time)

4.4.2 Performance Analysis of CIM-CS

The main purpose of CIM-CS is to incentivize clients to participate in training. Therefore, to validate the performance of CIM-CS, the number of clients participating in training is compared as a metric.

In this section, numerous experiments are carried out to evaluate how well the proposed incentive mechanism functions in federated learning. The quantity of participants produced by training the MNIST, Fashion MNIST, and CIFAR-10 datasets, respectively, is depicted in Figs. 12a–12c. The y-axis in this graph indicates how many clients participated in each communication round, while the x-axis lists the communication rounds. Figs. 12a–12c show that the CIM-CS stabilizes more quickly than the random mechanism. There are also more clients participating in the CIM-CS than in the random mechanism.



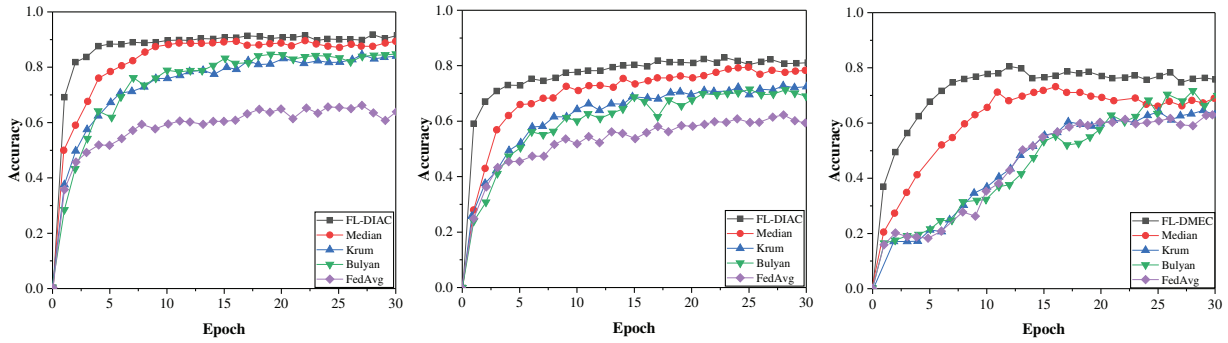
(a) MNIST (b) FashionMNIST (c) CIFAR-10

Figure 12: The number of clients per communication round selected by CIM-CS on three datasets ((a): MNIST, (b): Fashion MNIST, (c): CIFAR-10)

4.4.3 Performance Analysis of FL-DIAC

Similarly, we evaluate the FL-DIAC scheme performance by comparing the accuracy of each scheme.

We trained the MNIST dataset using the CNN model, the Fashion MNIST dataset using MLP, and the CIFAT-10 dataset using ResNet-18, respectively. From Figs. 13a–13c, we can see that the proposed FL-DIAC scheme is compared with the FedAvg, the Median scheme, the Krum scheme, and the Bulyan scheme. The accuracy of FL-DIAC, the Median scheme, the Krum scheme, and the Bulyan scheme are higher than the federated average, where FL-DIAC has the highest accuracy rate. The above results can directly indicate that the performance of FL-DIAC is expected to be better.



(a) MNIST (b) Fashion MNIST (c) CIFAR-10

Figure 13: Performance comparison with different schemes ((a): MNIST, (b): Fashion MNIST, (c): CIFAR-10)

In conclusion, we evaluate the proposed scheme on various benchmark datasets. Our findings demonstrate that in comparison to conventional schemes, our federated learning framework for Detecting and Incentivizing Abnormal Clients (FL-DIAC) scheme dramatically lowers the number of abnormal clients and enhances model performance. Additionally, our reward system successfully entices abnormal clients to take part in subsequent training sessions.

5 Conclusion

Federated learning is becoming increasingly important as the computational power of remote edge devices and local data privacy increase. However, issues with attacked clients in the federated learning framework still affect model accuracy. And for abnormal clients, previous work was chosen to be discarded, which will lead to the problem of decreasing model accuracy. Therefore, in this paper, we proposed a Federated Learning framework for Detecting and Incentivizing Abnormal Clients (FL-DIAC) to deal with the problem of identifying and incentivizing abnormal clients in federated learning. Specifically, on the one hand, an abnormal client detection model (DACM-D) was constructed. An autoencoder for abnormal detection was introduced in this model. In the federated learning framework, tens of thousands of clients are often involved during training. As a result, the range of parameters input to the autoencoder is substantial, which can lead to high computational complexity for abnormal client detection. In order to reduce the computational complexity, we also proposed a Fourier transform-based abnormal detection method for dimensionality reduction. On the other hand, the Incentive Model Based on Credit Score (CIM-CS) was constructed and this model was used to incentivize clients to attend the training in order to encourage the active participation

of abnormal clients. The experimental results show that the FL-DIAC scheme outperforms other algorithms of the same type in terms of model accuracy. The DACM-D model has higher accuracy than its counterparts under attacks of symbol flipping, additive noise, and gradient rise. DACM-D is more suitable for resource-constrained mobile edge environments. The Fourier transform-based anomaly data detection method proposed in this paper is effective. CIM-CS is efficient in incentivizing clients.

Although this paper solved some problems, due to the large number of clients, timely and fast-tracking of abnormal clients becomes a new problem. Future research should focus on introducing blockchain technology to secure federated learning and trace back anomalous clients.

Acknowledgement: The authors wish to express their appreciation to the reviewers and journal editors for their helpful suggestions which greatly improved the presentation of this paper. The authors are grateful for the support of Fundamental Research Funds for the Central Universities (No. B220203006).

Funding Statement: This work is supported by Key Research and Development Program of China (No. 2022YFC3005401), Key Research and Development Program of Yunnan Province, China (Nos. 202203AA080009, 202202AF080003), Science and Technology Achievement Transformation Program of Jiangsu Province, China (BA2021002), Fundamental Research Funds for the Central Universities (Nos. B220203006, B210203024).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Hongle Guo, Yingchi Mao; data collection: Xiaoming He; analysis and interpretation of results: Hongle Guo, Yingchi Mao, Ping Ping; draft manuscript preparation: Hongle Guo, Benteng Zhang, Tianfu Pang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: In this paper, we use the MNIST dataset. The URL of the dataset is http://yann.lecun.com/exdb/mnist/. We use the Fashion MNIST dataset. The URL of the dataset is https://github.com/zalandoresearch/fashion-mnist. We use the CIFAR-10 dataset. The URL of the dataset is http://www.cs.toronto.edu/kriz/cifar.html.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Guo, F., Yu, F. R., Zhang, H., Li, X., Ji, H. et al. (2021). Enabling massive iot toward 6G: A comprehensive survey. *IEEE Internet of Things Journal, 8(15),* 11891–11915.
2. Quach, S., Thaichon, P., Martin, K. D., Weaven, S., Palmatier, R. W. (2022). Digital technologies: Tensions in privacy and data. *Journal of the Academy of Marketing Science, 50(6),* 1299–1323.
3. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273–1282. Ft. Lauderdale, FL, USA, PMLR.
4. Guo, S., Zhang, T., Yu, H., Xie, X., Ma, L. et al. (2022). Byzantine-resilient decentralized stochastic gradient descent. *IEEE Transactions on Circuits and Systems for Video Technology, 32(6),* 4096–4106.
5. Shejwalkar, V., Houmansadr, A. (2021). Manipulating the byzantine: Optimizing model poisoning attacks and defenses for federated learning. *Proceedings of the 2021 Network and Distributed System Security Symposium*, San Diego, California, NDSS.

6. Wang, T., Zheng, Z., Lin, F. (2021). Federated learning framew ork based on trimmed mean aggregation rules. http://dx.doi.org/10.2139/ssrn.4181353
7. Li, L., Xu, W., Chen, T., Giannakis, G. B., Ling, Q. (2019). RSA: Byzantine-robust stochastic aggregation methods for distributed learning from heterogeneous datasets. *Proceedings of the AAAI Conference on Artificial Intelligence, 33(1),* 1544–1551.
8. Shen, S., Tople, S., Saxena, P. (2016). Auror: Defending against poisoning attacks in collaborative deep learning systems. *Proceedings of the 32nd Annual Conference on Computer Security Applications*, pp. 508–519. New York, USA, ACSAC.
9. Ding, N., Fang, Z., Huang, J. (2021). Optimal contract design for efficient federated learning with multi-dimensional private information. *IEEE Journal on Selected Areas in Communications, 39(1),* 186–200.
10. Wu, M., Ye, D., Ding, J., Guo, Y., Yu, R. et al. (2021). Incentivizing differentially private federated learning: A multidimensional contract approach. *IEEE Internet of Things Journal, 8(13),* 10639–10651.
11. Sarikaya, Y., Ercetin, O. (2020). Motivating workers in federated learning: A stackelberg game perspective. *IEEE Networking Letters, 2(1),* 23–27.
12. Ding, N., Fang, Z., Huang, J. (2020). Incentive mechanism design for federated learning with multi-dimensional private information. *Proceedings of the 2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT)*, pp. 1–8. Volos, Greece, IEEE.
13. Feng, S., Niyato, D., Wang, P., Kim, D. I., Liang, Y. -C. (2019). Joint service pricing and cooperative relay communication for federated learning. *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 815–820. Atlanta, GA, USA, IEEE.
14. Zeng, R., Zhang, S., Wang, J., Chu, X. (2020). Fmore: An incentive scheme of multi-dimensional auction for federated learning in MEC. *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, pp. 278–288. Singapore, IEEE.
15. Lim, W. Y. B., Xiong, Z., Miao, C., Niyato, D., Yang, Q. et al. (2020). Hierarchical incentive mechanism design for federated machine learning in mobile networks. *IEEE Internet of Things Journal, 7(10),* 9575–9588.
16. Zhan, Y., Li, P., Wang, K., Guo, S., Xia, Y. (2020). Big data analytics by crowdlearning: Architecture and mechanism design. *IEEE Network, 34(3),* 143–147.
17. Zhan, Y., Zhang, J., Li, P., Xia, Y. (2019). Crowdtraining: Architecture and incentive mechanism for deep learning training in the Internet of Things. *IEEE Network, 33(5),* 89–95.
18. Zhan, Y., Li, P., Qu, Z., Zeng, D., Guo, S. (2020). A learning-based incentive mechanism for federated learning. *IEEE Internet of Things Journal, 7(7),* 6360–6368.
19. Zhan, Y., Xia, Y., Zhang, J. (2018). Incentive mechanism in platform-centric mobile crowdsensing: A one-to-many bargaining approach. *Computer Networks, 132,* 40–52.
20. Li, Y., Li, F., Zhu, L., Sharif, K., Chen, H. (2022). A two-tiered incentive mechanism design for federated crowd sensing. In: *CCF transactions on pervasive computing and interaction*, vol. 4, pp. 339–356. Springer.
21. Zhan, Y., Zhang, J., Hong, Z., Wu, L., Li, P. et al. (2022). A survey of incentive mechanism design for federated learning. *IEEE Transactions on Emerging Topics in Computing, 10(2),* 1035–1044.
22. Mills, J., Hu, J., Min, G. (2022). Client-side optimization strategies for communication-efficient federated learning. *IEEE Communications Magazine, 60(7),* 60–66.
23. Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A. et al. (2022). Federated-learning-based anomaly detection for iot security attacks. *IEEE Internet of Things Journal, 9(4),* 2545–2554.
24. Salahuddin, M. A., Faizul Bari, M., Alameddine, H., Pourahmadi, A., Boutaba, V. (2020). Time-based anomaly detection using autoencoder. *2020 16th International Conference on Network and Service Conference on Network and Service Management (CNSM)*, pp. 1–9. Izmir, Turkey, IEEE.

25. Kim, S., Jo, W., Shon, T. (2020). APAD: Autoencoder-based payload anomaly detection for industrial IoE. *Applied Soft Computing, 88,* 106017.
26. Badrinarayanan, V., Kendall, A., Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(12),* 2481–2495.
27. Yamada, Y., Morimura, T. (2016). Weight features for predicting future model performance of deep neural networks. *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*, pp. 2231–2237. New York, USA.
28. Ghosh, A., Chung, J., Yin, D., Ramchandran, K. (2022). An efficient framework for clustered federated learning. *IEEE Transactions on Information Theory, 68(12),* 8076–8091.
29. Wang, H., Zhu, X., Chen, P., Yang, Y., Ma, C. et al. (2022). A gradient-based automatic optimization CNN framework for EEG state recognition. *Journal of Neural Engineering, 19(1),* 016009.
30. Li, J., Liu, R., Yu, R., Wang, X., Zhao, Z. (2014). Reputation-based participant incentive approach in opportunistic cognitive networks. *Advanced Computer Architecture*, pp. 201–214. Berlin Heidelberg, Springer.
31. Alzubi, J. A., Alzubi, O. A., Singh, A., Ramachandran, M. (2023). Cloud-IIoT-based electronic health record privacy-preserving by CNN and blockchain-enabled federated learning. *IEEE Transactions on Industrial Informatics, 19(1),* 1080–1087.
32. Zhu, H., Jin, Y. (2020). Multi-objective evolutionary federated learning. *IEEE Transactions on Neural Networks and Learning Systems, 31(4),* 1310–1322.
33. He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. Las Vegas, USA, IEEE.
34. LeCun, Y. (1998). The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/ (accessed on 11/03/2023)
35. Kayed, M., Anter, A., Mohamed, H. (2020). Classification of garments from fashion MNIST dataset using CNN lenet-5 architecture. *2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE)*, pp. 238–243. Aswan, Egypt, IEEE.
36. Krizhevsky, A., Hinton, G. (2009). Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases, 1(4),* pp. 1–54.
37. Chen, Y., Su, L., Xu, J. (2018). Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *ACM SIGMETRICS Performance Evaluation Review, 46(1),* 96.
38. Guerraoui, R., Rouault, S. (2018). The hidden vulnerability of distributed learning in byzantium. *International Conference on Machine Learning*, pp. 3521–3530. Stockholm, Sweden, PMLR.