Neurocomputing 574 (2024) 127227

Contents lists available at ScienceDirect

Neurocomputing

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

FedAVE: Adaptive data value evaluation framework for collaborative fairness in federated learning

Zihui Wang [a,b,c], Zhaopeng Peng [a,b,c], Xiaoliang Fan [a,b,c,*], Zheng Wang [a,b,c], Shangbin Wu [a], Rongshan Yu [a,b,c], Peizhen Yang [a,b,c], Chuanpan Zheng [a], Cheng Wang [a,b,c]

[a] Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, 361005, PR China
[b] National Institute for Data Science in Health and Medicine, Xiamen University, PR China
[c] Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, 361005, PR China

ARTICLE INFO

Communicated by J. Andreu-Perez

Keywords:
Federated learning
Collaborative fairness
Reputation

ABSTRACT

Collaborative fairness in federated learning rewards high-contribution clients with high-performance models when multiple clients train a machine learning model cooperatively. Existing approaches ignore the information on data distribution when evaluating the clients' data quality, resulting in a mismatch between the reward allocation and the real data quality of clients under different data heterogeneity settings. To address this problem, we propose a novel Federated learning framework with Adaptive data Value Evaluation mechanism (FedAVE) to ensure collaborative fairness without affecting the predictive performance of models. First, an *adaptive reputation calculation module* is designed to generate reputations that match the clients' contributions based on the information of their data distribution, respectively. Second, a *dynamic gradient reward distribution module* is devised to allocate a certain number of aggregated model parameter updates/gradients as the rewards corresponding to the reputations and the data distribution information. Extensive experiments on three public benchmarks show that the proposed FedAVE outperforms all baseline methods in terms of fairness, and achieves comparable performance to the state-of-the-art methods in terms of accuracy. Code available at https://github.com/wangzihuixmu/FedAVE.

1. Introduction

Federated Learning (FL) is a promising approach of large-scale collaborative learning that allows clients to train a global model together while preserving the local data privacy [1–3]. As the global model is expected to outperform the locally trained model, FL attracts wide attention in different applications, including healthcare, criminal justice, etc [4–6].

Currently, most FL methods [1–3,7] distribute the same model to all clients in each communication round without considering their contributions to the system. These approaches tend to discourage high-quality clients from actively participating in FL. [8–11]. As shown in Fig. 1, the contribution of each client is different to the system because they have different data sizes and distributions, such as $Client_1$ and $Client_2$. The framework should distribute different models (more corners denote more rewards) based on their contributions (Ground truth) to make sure collaborative fairness. In this way, clients with high contributions are willing to join FL.

In FL, existing collaborative fairness methods include two steps [11,12]: contribution evaluation (i.e., reputation) and reward allocation [13,14]. Generally, there are three methods for achieving collaborative fairness in FL: the similarity-based method (CGSV [13]), the data size-based method (CFFL (a) [14]), and the diversity of labels-based method (CFFL (b) [14]). CGSV recognizes higher-contribution clients as those whose gradients are more similar to the averaging gradient than others. Subsequently, they are rewarded with more amounts (i.e., the hyperbolic tangent of the contribution) of gradients. CFFL recognizes higher-contribution clients as those whose local data and local models are both of higher quality than others (i.e., larger data sizes in CFFL (a) or more diverse labels of local data in CFFL (b) for the local data, and better performance on the validation data for the local models). Then, they are rewarded with more amounts (i.e., the ratio of data quality multiplying the hyperbolic sine of the contribution) of gradients. In summary, the two methods (i.e., CGSV and CFFL) fail to enhance collaborative fairness when data heterogeneity varies across clients. We summarize that the major limitations of state-of-the-art methods are in twofold:

* Corresponding author at: Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, 361005, PR China.
E-mail addresses: wangziwei@stu.xmu.edu.cn (Z. Wang), fanxiaoliang@xmu.edu.cn (X. Fan).

https://doi.org/10.1016/j.neucom.2023.127227
Received 24 August 2022; Received in revised form 26 December 2023; Accepted 30 December 2023
Available online 5 January 2024

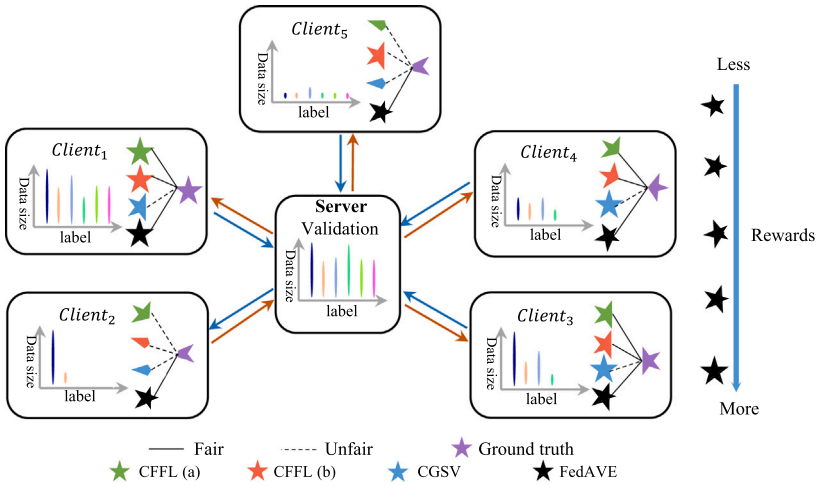

Fig. 1. Collaborative fairness in FL. We compare the FedAVE with three typical methods (e.g. CFFL (a), CFFL (b) and CGSV) which fail to ensure collaborative fairness under different data heterogeneity settings because of ignoring data distribution information of clients. The stars with different colors represent four methods and more corners denote more rewards. The closer to the ground truth, the better the fairness.

(1) In the contribution evaluation phase, existing works fail to fairly evaluate the reputations of different clients when the data heterogeneity varies. For example, in Fig. 1, the reputations of $Client_3$ and $Client_4$ are over-estimated by CGSV, as their gradients are more similar to the majority of clients' gradients. CFFL cannot accurately estimate the ground-truth reputations for $Client_5$ and $Client_2$ simultaneously, as it merely uses either the local data sizes or the label diversity to evaluate the data quality. (2) In the reward allocation phase, the challenge lies in determining which parameters of the gradient to allocate to clients, given the diverse contributions of each parameter of the gradient to the model's performance [15–17]. For instance, small differences in the amounts of the gradients allocated to clients may cause similar rewards (i.e., model performance) among clients, leading to unfair treatments to clients with higher contributions. In summary, the challenge of achieving collaborative fairness in FL has not been fully tackled when data heterogeneity varies across clients.

To address these problems, we propose a novel **Fed**erated learning framework with **A**daptive data **V**alue **E**valuation mechanism (**FedAVE**) to ensure collaborative fairness with obtaining competitive predictive accuracy. FedAVE contains two modules: the adaptive reputation calculation module and the dynamic gradient reward distribution module. The first module is *the adaptive reputation calculation module* that is designed to calculate clients' reputations in each round by two aspects: the performance of the local model on the validation set stored in the server, which partially reflects the information of the local dataset; and the Kullback–Leibler (KL) divergence to model the difference between the local data and validation set. The module treats clients with high similarity between local dataset and the validation set as high-contributor clients. In *the dynamic gradient reward distribution module*, a certain number of aggregated model parameter gradients are allocated as rewards based on the reputations and the data distribution information, ensuring that the rewards obtained by them have distinction significant. This mechanism guarantees fairness in the form of the model performance.

In summary, the main contributions of this paper include the following:

- We propose a novel federated learning framework FedAVE to ensure collaborative fairness without affecting the performance of models under different data heterogeneity settings (i.e., the data sizes and distributions are different simultaneously among clients).
- In the contribution evaluation phase, we design an adaptive reputation evaluation module, which fairly and accurately estimates the contributions for different clients based on their local data distribution information.
- In the reward allocation phase, we conduct a dynamic gradient reward allocation module, which significantly distinguishes the rewards (i.e., the parameters of the gradients) to clients according to their estimated contributions in each round to enhance collaborative fairness.
- Experimental results on three popular federated benchmarks show that the proposed FedAVE outperforms all baseline methods in terms of fairness, and achieves comparable performance to the state-of-the-art methods in terms of accuracy under different data heterogeneity settings.

2. Related work

In FL, designing appropriate rewards to facilitate collaboration among different clients is a meaningful and essential question [1,3,12,18,19]. A well-designed reward mechanism should contain a fairness standard, a proper reward form, and a systematic method to ensure fairness. In the domain of fairness within FL, we have conducted a review of related works to better comprehend our study in comparison to existing research.

Incentive mechanism. Some researchers design the incentive mechanism to reward clients with monetary or the total revenue generated collaboratively. Yu et al. [20] dynamically allocate rewards to clients in a context-aware manner; consequently, the reduction of unfairness among clients is achieved when the maximum utility is attained. Zhang et al. [21] propose an incentive mechanism based on the reputation and reverses the auction theory to reward clients by combining their reputations with a limited budget. Although it is natural to consider monetary incentives [20,22,23], these methods are difficult to implement as the value between models/datas and money is hard to balance [24,25].

Egalitarian fairness. Another research direction involves egalitarian notions of fairness, where all clients receive the global model that performs equally on their local datasets [26]. q-FFL [27] proposes a modified local loss function that provides higher optimization weights for clients with higher loss. FedFV [28] identifies gradient conflicts with large differences as a cause of unfairness in FL, and mitigates potential conflicts between clients before averaging gradients. It is worth noting that all the clients download the same global model regardless of their contributions to the system.

Collaborative fairness. On the contrary, recent works focus on the collaborative fairness of FL, which treats the global model as rewards for clients, excepting the models received by the clients matched their contributions. FPPDL [12] proposes a method for the mutual evaluation

of the local credibility mechanisms to guarantee the fairness (i.e., each client privately evaluates other clients). Since the framework does not have a server, resulting in it is not applied to FL. CGSV [13] proposes a method based on the cosine gradient Shapley value, calculated by assessing the similarity between the client's gradients and the overall gradients, to compute clients' reputations. Subsequently, this approach utilizes the obtained reputations for the distribution of rewards. However, the method is hardly applied to the different data heterogeneity settings, e.g., the data sizes and distributions are different simultaneously among clients. CFFL [14] achieves collaborative fairness by appending a validation set, and the reputations are computed from the diversity of labels (or data sizes) of the clients, along with the validation performance of the local model to allocate rewards. Whereas, it is not suitable for the scenarios where the data heterogeneity varies across clients. Different from these works, the proposed FedAVE calculates the clients' reputations through the local models' performance and the data distribution of its dataset, which accurately reflect their similarity to the validation set. This allows us to successfully apply it to the different data heterogeneity settings.

3. Preliminaries

3.1. Federated learning

A FL system consists of m clients and a server node that aggregates collected models. The goal is to train a global model (ω) that minimizes the weighted average loss of all clients without requiring them to upload their private local data, e.g., FedAvg [2]. First, the server randomly initializes a global model, ω_0, and distributes it to each client. Then, models are sent to the server for aggregation after training for E number of epochs iteratively. Finally, the server broadcasts the aggregation model to available clients for the next communication round. These steps are repeated for a total of T times until the global model is converged. Eq. (1) shows the traditional objective function of FL:

$$\min_{\omega} F(\omega) = \sum_{i=1}^{m} p_i F_i(\omega), \tag{1}$$

where ω is the global model of the aggregation, i denotes the ith client and m is the total number of clients. The local objective function of ith client with weight p_i is denoted by $F_i(\omega)$, where $p_i \geq 0$ and $\sum_{i=1}^{m} p_i = 1$. To minimize the weighted average loss of all clients, FedAvg randomly samples a subset S_t of m clients, $0 < \mathrm{i} \leq m$, to update the global model at communication round t:

$$\omega^{t+1} = \frac{1}{|S_t|} \sum_{i \in S_t} p_i \omega_i^t, \tag{2}$$

where ω^{t+1} denotes the global model at communication round $t+1$, $p_i = \frac{n_i}{\sum_{j=1}^{m} n_j}$ denotes the weight of ith client and n_i is the data sizes of the client i. FedAvg has been proved to be efficient in minimizing the objective while protecting the privacy. However, it may be unfair to the high-quality clients [13,14].

3.2. Collaborative fairness

The key of reward designation is that clients who contribute more will receive more rewards [9,29]. Different from other rewards, we design the reward corresponding to its model's performance. The Pearson Correlation Coefficient is first used by [13,14] to measure the correlation between clients' contributions and the reward they are received. It will be close to 1 when the client's reward is positively correlated with their contribution (i.e., client who contributes more are allocated more rewards), and it will be close to −1 when the correlation is negative. Therefore the Pearson Correlation Coefficient can reflect the degree of collaborative fairness. The definition of collaborative fairness given as follows:

Definition 1 (*Collaborative Fairness*)**.** Clients' contribution (x) is calculated by comparing the performance of standalone models (without collaboration) and the performance of the final models (y) obtained by the clients after collaborating. The quantitative fairness computed by $\gamma := 100 \times \rho(x, y)$ where $\rho()$ is the Pearson Correlation Coefficient. The γ represents the degree of linear correlation between x and y. The higher the connection between x and y, the closer γ is to 100, and the better fairness of the framework.

3.3. Quantification of fairness

To evaluate fairness reasonably, we follow [14] to quantify the collaborative fairness with Pearson Correlation Coefficient $\gamma := 100 \times \rho(x, y) \in [-100, 100]$ between client contributions (x: test accuracies of standalone models which are optimized by their own local datasets) and client rewards (y: test accuracies of local models after collaboration).

Agent with higher standalone accuracies indicates contribute more. x can be written into Eq. (3), where $sacc_i$ represents the performance of the client i's standalone model:

$$x = \{sacc_1, sacc_2, sacc_3, \ldots, sacc_m\}, \tag{3}$$

y can be written into Eq. (4), where acc_j denotes the performance of the client j's local model after collaborating:

$$y = \{acc_1, acc_2, acc_3, \ldots, acc_m\}. \tag{4}$$

Finally, we quantify the collaboration fairness γ by Eq. (5):

$$\gamma_{xy} = 100 \times \frac{\sum_{i=1}^{m}(x_i - \bar{x})(y_i - \bar{y})}{(m-1)s_x s_y}, \tag{5}$$

where $\bar{x}$ and $\bar{y}$ denote the mean of x and y, s_x and s_y denote the standard deviation of x and y. The range of values of γ is $[-100, 100]$, within higher γ implying better fairness of framework.

4. Methodology

The key of achieving collaborative fairness is to reasonably allocate the model according to their contributions. In particular, we manage the gradients downloading of clients based on their reputations to ensure the system fairness. The reputations are managed by the server and invisible to clients. In this section, we will introduce the workflow and architecture of the FedAVE, which mainly consists the *Adaptive Reputation Calculation (ARC) module and the Dynamic Gradients Reward distribution (DGR) module*.

4.1. Overview

Fig. 2 shows an overview of the proposed FedAVE framework. After randomly assigning a global model to the clients, a round of FedAVE communication consists of the following steps: client sampling, global aggregation, client reputation calculation, and reward distribution.

Step 1 Since our scene contains a few clients, we use full sampling to ensure all the clients joining FL in each round.

Step 2 Global aggregation step is used to generate a new global model with the gradients of the clients' model.

Step 3 Measuring the clients' reputations in each round based on the similarity between the clients' dataset and the validation set.

Step 4 The server distributes the aggregated model parameter gradients to clients corresponding to their reputations.

The following sections will discuss the detailed procedures and theoretical basis of the FedAVE, as summarized in Algorithm 1.

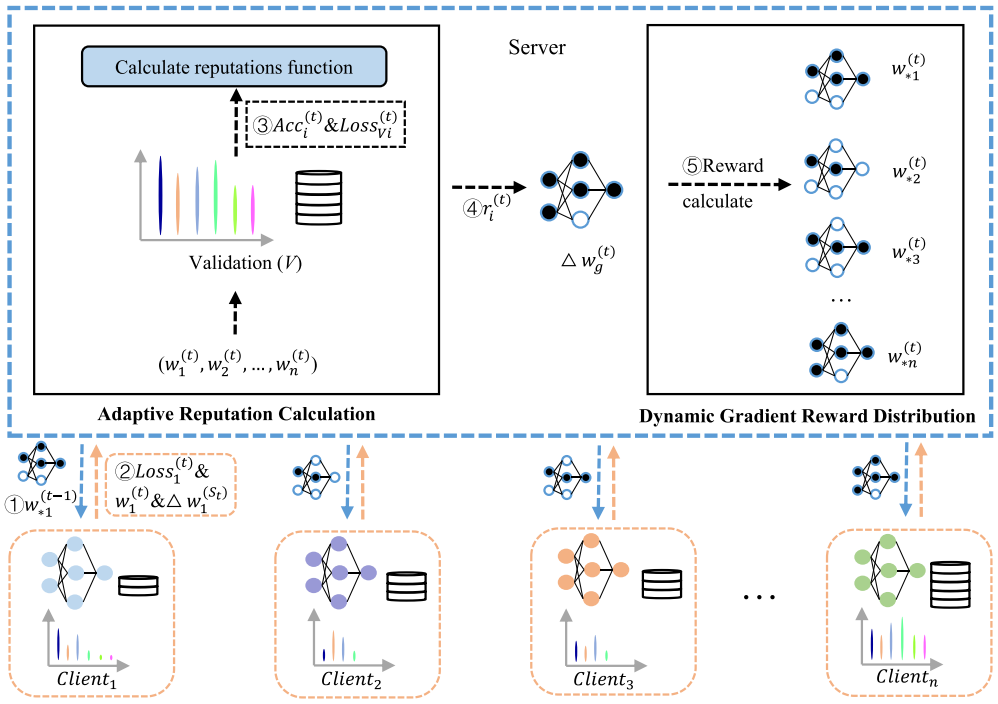

Fig. 2. The FedAVE framework. The framework consists of two modules: (1) Adaptive Reputation Calculation module to compute the reputations of the clients with different contributions; (2) Dynamic Gradients Reward Distribution module to distribute model rewards based on the reputations.

Algorithm 1 FedAVE

Input: local epochs E, batch size B, number of clients m, data sizes owned by each client n_i, validation set V, gradient normalizing constant γ, hyper-parameter β, hyperparameter α, model parameters vector dimension D.

Client i

1: Local gradients: $\Delta w_i^{(t)} := \nabla F_i(w_i^{(t-1)})$
2: Clips gradients vector: $\Delta w_i^{(t)} := clip(\Delta w_i^t)$
3: Send the gradients $\Delta(w_i^{S_t}) = \Delta(w_i^{(t)}) * \tau / ||\Delta(w_i^t)||$ to the server in round t
4: Download reward gradients based on the "largest values" criteria $\Delta w_{*i}^{(t)}$, and integrate with local model: $w_i^{(t+1)} = w_i^{(t)} + \Delta w_{*i}^{(t)}$

Server

Aggregation:
5: $\Delta w_g^t = \sum_{i=1}^{m} \frac{n_i}{\sum_{i=1}^{m} n_i} * \Delta(w_i^{S_t})$
Calculate the reputation of client i in round t:
6: **for** $i \in R$ **do**
7: $Acc_i^{(t)} = V(w_i^{(t)} + \Delta w_i^{(t)})$
8: $\tilde{r}_i^{(t)} = Acc_i^{(t)} / KL(Loss_i^{(t)}, Loss_{Vi}^{(t)})$
9: $r_i^{(t)} = \alpha * r_i^{(t-1)} + (1-\alpha) * \tilde{r}_i^{(t)}$
10: *Normalized reputation:* $r_i^{(t)} = \frac{r_i^{(t)}}{\sum_{i=1}^{m} r_i^{(t)}}$
11: **end for**
Distribute rewards based on their reputations:
12: **for** $i, j \in R$ **do**
13: $quota_i^t := [D \times tanh(\beta r_i^{(t)}) / (max_{j \in N} tanh(\beta r_j^{(t)}) * KL(Loss_i^{(t)}, Loss_{Vi}^{(t)}))]$
14: *Reward of client i:*
15: $\Delta w_{*i}^{(t)} = sparsify(\Delta w_g^{(t)}, quota_i^t)$
16: **end for**

4.2. FedAVE

To improve the performance of the global model, it is necessary to make full use of the information of clients. However, existing collaborative fairness methods all tend to favor high-contributor clients when aggregation. As a result, the data from low-contributors are underrepresented by the models, leading to a decrease in the overall quality of the models. To make the best use of all the local data, we employ Eqs. (6) and (7) to calculate the aggregated model in round t as follows, denoted as Δw_g^t:

$$\Delta(w_i^{S_t}) = \Delta w_i^{(t)} * \tau / \|\Delta w_i^{(t)}\|, \tag{6}$$

$$\Delta w_g^t = \sum_{i=1}^{m} \frac{n_i}{\sum_{i=1}^{m} n_i} * \Delta(w_i^{S_t}), \tag{7}$$

where $\Delta w_i^{(t)} := \nabla F_i(w_i^{(t-1)})$ is the model gradient updated by client i, n_i is the data sizes of the client i, $\Delta(w_i^{S_t})$ is the gradient that the client i has uploaded to the server, and τ is a normalization coefficient used to prevent the gradient explosion [30,31]. In FL, the model parameters uploaded by clients may be quite different. Based on the normalization of the gradients of clients' uploads, Eq. (6) can prevent the aggregation model from being dominated by a single one, thereby fostering the development of the system [13,19].

ARC module. Next, we present the technical details of the ARC module, which employs data distribution more effectively than CFFL in calculating the reputations. This provides it with a distinct advantage under different data heterogeneity settings. Inspired by FedKD [32], we propose the module to calculate the reputations of clients through the distribution of the loss value, which reflects part of the information in the dataset. We assumed the validation set to be standard in the server (i.e., the same data distribution to the union of all the local data). For example, the loss distribution value of testing on the local dataset and the validation set are closer, which means the quality of clients' data are similar to the validation set. Then, we regard them as high-contribution clients. Eqs. (8) and (9) formulate the reputation $r_i^{(t)}$ of client i in round t:

$$\tilde{r}_i^{(t)} = Acc_i^{(t)} / KL(Loss_i^{(t)}, Loss_{Vi}^{(t)}), \tag{8}$$

$$r_i^{(t)} = \alpha * r_i^{(t-1)} + (1-\alpha) * \tilde{r}_i^{(t)}, r_i^{(t)} = \frac{r_i^{(t)}}{\sum_{i=1}^{m} r_i^{(t)}}, \tag{9}$$

Table 1
Main notations.

Notations	Description
m	Total number of clients
Δw_g^t	The global model in round t
w_i^t	Client i's model in round t
Δw_i^t	Client i's gradients in round t
$\Delta w_{*i}^{(t)}$	The reward gradients downloaded by client i from the server
$\Delta w_i^{(S_t)}$	Gradients uploaded by client i to the server in round t
F_i	Loss functions of client i
$clip()$	Clipping the model parameters
n_i	Data sizes of client i
V	Validation set
τ	Gradient normalizing constant
β	Hyperparameter
$Acc_i^{(t)}$	The performance of the client i's model on the validation set in round t
$r_i^{(t)}$	Client i's reputation in round t
$Loss_i$	The distribution of loss values for client i's model tested on i's dataset
$Loss_{Vi}$	The distribution of loss values for client i's model tested on V
tanh	Hyperbolic tangent function
D	Model parameters vector dimension
sparsify	A function to distribute rewards based on client reputation

where $Acc_i^{(t)}$ represents the performance of the model i on the validation set, $Loss_i^{(t)}$ and $Loss_{Vi}^{(t)}$ denote the loss distribution value of the model i test on the client i's dataset and the validation set, respectively; $\tilde{r}_i^{(t)}$ is i's reputation in round t and α is an adaptive weight. In Eq. (9), we update the reputations based on the current round and the previous round. Meanwhile, the reputations computed smoothly without abrupt fluctuations and eliminate the noise generated during the training process.

DGR module. For most of the methods, clients obtain the same aggregated gradients/model from the server in the distribution step, which leads to clients expect better prediction performance [33,34]. However, it is unfair and discourages the clients with high quality from joining FL [14]. To guarantee fairness, the server should allocate the corresponding aggregated model parameter gradients as rewards based on the clients' contributions. Whereas, existing methods distribute rewards by the computed reputations simply, such as CFFL. To improve fairness, the DRG module assigns the corresponding parameter gradients $\Delta w_{*i}^{(t)}$ to client i corresponding to $r_i^{(t)}$ as follows,

$$quota_i^t := D \times tanh(\beta r_i^{(t)}) / (max_{j \in N} tanh(\beta r_j^{(t)}) * KL(Loss_i^{(t)}, Loss_{Vi}^{(t)})), \quad (10)$$

$$\Delta w_{*i}^{(t)} = \text{sparsify}(\Delta w_g^{(t)}, quota_i^t), \quad (11)$$

where D is model parameters vector dimension, $quota_i^t$ is the number of model parameters distributed by the server to the clients and determined by the relative reputations, β is a hyper-parameter. Sparsifying gradient vectors denotes that each node sorts gradients by the magnitude of the weights and only reward a subset of the component based on the reputations [35,36]. After sorting the gradients, $\text{sparsify}(\Delta w_g^{(t)}, quota_i^{(t)})$ keeps the largest $max(0, quota_i^{(t)})$ component in $\Delta w_g^{(t)}$ and zeros out all of its other components. The quality of the allocated model should be properly preserved to avoid model divergence during the training stage [17]. Considering earlier studies [15,16] that have shown a strong correlation between a model's weight magnitude and its importance for model quality, we employ a parameter-cutting and allocation strategy based on weight magnitudes. This approach not only facilitates reward allocations but also avoids model divergence. The notations are summarized in Table 1.

5. Experiment and discussion

5.1. Experimental settings

Dataset. We implement experiments on three benchmark datasets, MNIST [37], CIFAR-10 [38] and EMNIST letters [39]. MNIST contains 60,000 training and 10,000 testing images, whose sizes are 28 by 28 pixels, with labels ranging from 0 to 9. CIFAR-10 is a standard image classification dataset of size 32 by 32 pixels and contains 50,000 training and 10,000 testing images (1000 images per label) from 10 different labels. EMNIST letters is a 28 by 28 pixels classification dataset that includes 128,000 training and 20,800 test images from 26 distinct labels.

Baseline. We compare the FedAVE with the following state-of-the-art methods:

- FedAvg [2] is currently the most popular framework and it will distribute the same rewards to clients.
- CFFL [14] achieves collaborative fairness by appending a validation set, to fairly distribute rewards, reputations are computed from the diversity of labels (or data sizes) of the clients and the validation performance of the local model.
- CGSV [13] achieves collaborative fairness based on cosine gradient Shapley value that is calculated by the similarity between the clients' gradients and the overall gradients to evaluate their contributions and use them as reputations to distribute rewards.
- A Standalone framework where clients train local models alone without collaborative.

Data splits. We construct three heterogeneous settings by the clients' data sizes and distributions. For **imbalance data sizes (POW)** [13], we follow a power law to make the clients have different data sizes with the same classes, where a larger local data size suggests a higher similarity to the server validation set. For **imbalanced class numbers (CLA)** [13], we exchange the number of classes, while keeping their data sizes at 600 (MNIST)/2000 (CIFAR-10). For example, clients 1, 2, 3, 4 and 5 respectively own datasets with 1, 3, 5, 7 and 10 classes, and the similarity of their data to the server validation set increases with the number of classes owned by them. For **imbalanced data sizes and class numbers (DIR)**, we follow [40–42] to construct real-world statistical heterogeneity with Dirichlet distribution that makes clients have different data sizes and class numbers, where a more standard data distribution (i.e., a larger data size and a more diverse distribution of classes in data) suggests a higher similarity to the server validation set. In particular, we sample $p_i^l \sim Dir(\beta')$ and allocate a proportion of p_i^l of the data to client i with class l, where $Dir(\beta')$ is the Dirichlet distribution with a parameter of β'. In FL, the validation set, assumed to be IID (independent and identically distributed), can be stored on the server, but its size is so limited that it cannot be independently used to train a useful model. Therefore, rewarding participants differently is needed to enhance collaborative fairness, which motivates the participants to contribute the local data to improve the quality of the global model, especially in a competitive environment [13,14]. To compare with CFFL fairly, we constructed the validation set by splitting 10% evenly distributed data from the original training set randomly as same as CFFL [14].

Hyper-Parameters. On the MNIST dataset, we use a 2-layer convolutional neural network (CNN) [43]. The hyperparameters are: batch size $B = 32$, learning rate $lr = 0.15$ for $P = 10$, local epochs $E = 3$, gradient clipping between $[-0.001, 0.001]$, the moving average coefficient $\alpha = 0.95$, the gradient normalizing constant $\gamma = 0.5$, hyperparameter $\beta = 1.5$. On the CIFAR-10 dataset, we employ a 3-layer CNN [44]. The hyperparameters are: batch size $B = 128$, learning rate $lr = 0.015$ for $P = 10$, local epochs $E = 3$, gradient clipping between $[-0.001, 0.001]$, the moving average coefficient $\alpha = 0.95$, the gradient normalizing

Table 2
Fairness results (%) of different frameworks in various heterogeneous settings. A higher value means better fairness.

Dataset	MNIST					CIFAR-10					EMNIST letters				
No. agents	10					10					10				
Data partition	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)
FedAvg [2]	49.47	64.17	20.73	6.52	50.68	−20.66	88.92	−30.73	86.88	79.02	−4.18	71.61	23.38	37.23	13.72
CFFL [14]	95.11	**99.82**	–	–	–	79.95	**99.74**	–	–	–	74.98	87.43	–	–	–
CGSV [13]	91.20	92.32	66.81	71.54	93.29	95.48	95.46	**77.45**	80.33	71.79	**99.21**	79.76	51.34	51.81	38.45
Ours	**97.59**	97.79	**93.62**	**84.22**	**94.78**	**99.33**	97.89	74.40	**98.36**	**91.87**	95.48	**89.77**	**81.24**	**75.57**	**69.69**

Table 3
The maximum test accuracy (%) over three public benchmarks achieved by different baselines.

Dataset	MNIST					CIFAR-10					EMNIST letters				
No. agents	10					10					10				
Data partition	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)
Standalone	94.79	92.19	65.17	71.64	85.73	48.66	43.13	30.80	33.25	38.03	88.07	85.14	43.94	55.62	70.42
FedAvg [2]	**97.67**	**95.87**	**98.74**	**98.76**	**98.94**	**59.94**	**52.40**	**60.77**	**61.23**	**62.53**	**92.26**	**90.24**	**89.79**	**89.33**	**89.93**
CFFL [14]	91.78	88.35	–	–	–	44.20	48.66	–	–	–	86.87	81.72	–	–	–
CGSV [13]	96.00	91.40	96.09	98.17	98.80	58.13	41.49	33.35	44.86	49.31	91.6	89.77	87.68	88.41	87.8
Ours	97.60	93.23	96.59	**98.76**	98.74	59.91	50.82	53.78	55.99	54.42	92.06	90.07	88.19	87.07	89.4

constant $\gamma = 0.15$, hyperparameter $\beta = 1.5$. On the EMNIST letters dataset, we employ the ResNet18 network [45]. The hyperparameters are: batch size $B = 128$, learning rate $lr = 0.15$ for $P = 10$, local epochs $E = 1$, gradient clipping between $[-0.001, 0.001]$, the moving average coefficient $\alpha = 0.95$, the gradient normalizing constant $\gamma = 0.15$, hyperparameter $\beta = 1.5$.

5.2. Experiment results

We evaluate the validity of the FedAVE on two metrics: (1) fairness; (2) the maximum predictive performance of the client's model.

Fairness comparison. To verify the effectiveness of FedAVE, we compare a few algorithms in the different heterogeneous settings under the number of 10 clients. Table 2 shows the results of fairness results (the Pearson Correlation Coefficient between the standalone performance and the final model performance) achieved by the baselines on three datasets. The standalone performance remains unaffected by the methods employed. In our comparison, it is used to represent the client's contribution, facilitating a fair evaluation of results across different methods. Table 2 shows that our method achieves high fairness of over 84%, while the commonly used FedAvg whose minimum fairness is 6.52% performs not well. For the CLA data partition on MNIST, CFFL outperforms our method by 2.03% in fairness. The reason is that the total number of the diversity of labels reflect its reputation accurately at this time, while the method of CFFL is not suitable for DIR. For the DIR (0.1) data partition on CIFAR-10, although CGSV outperforms our method by 3.05% in fairness, we achieve an overall accuracy improvement of +20.43%. This demonstrates the superiority of our method over CGSV. Table 2 indicates that the proposed FedAVE compares favorably against the state-of-the-art methods in terms of fairness, which means that our notion of fairness: a high-contribution client can get the model with better performance. We also observe that FedAvg always get lower fairness than FedAVE, which is obvious as they do not consider the concept of fairness well in the framework.

Predictive performance. Table 3 reports the best accuracy achieved among clients. Due to our FedAVE, which allows clients to obtain different final models, we expect that the client contributes the most to have the maximum reward. It can be found that FedAVE can get comparable accuracy to FedAvg, and consistently surpass the Standalone. For example, for CLA data partition on MNIST, we observe that our result obtains 93.23%, which is higher than Standalone (92.19%) and CGSV (91.40%), and slightly lower than FedAvg (95.87%). In particular, our method achieves the same maximum accuracy compared with FedAvg in DIR (0.2) data partition on MNIST. Fig. 3 shows the clients' test accuracy changes with the communication rounds increase on MNIST and CIFAR-10 in different scenarios. As the data sizes and label categories owned by clients vary in FL, they contribute differently to the system. The proposed FedAVE calculates corresponding reputations based on the clients' contributions, and then dynamically allocates rewards according to the reputations. Finally, each client will converge to a different model and eventually get different test accuracy. Specifically, for CLA data partition on CIFAR10, the model's performance of the low-quality clients decline as the communication rounds increasing. Since the significant difference in contributions among the clients in CLA data partition, the rewards of some low-quality clients received are fixed and lower, which changes the preference of the models and decreases the model's performance. Fig. 4 shows the distributions of the ground-truth contributions (i.e., the blue bars of Standalone) and the allocated rewards by FedAVE (i.e., the orange bars of FedAVE) under three cases (i.e., CLA in MNIST, CIFAR10 and EMNIST letters). The consistency between the contributions and the rewards confirms the ability of our proposed FedAVE in enhancing collaborative fairness.

Table 4
Ablation studies of proposed method on MNIST. Fairness results (%) of the FedAVE.

Dataset	MNIST				
No. agents	10				
Data partition	POW	CLA	DIR (0.1)	DIR (0.2)	DIR (0.3)
FedAVE$\|r^-q^-$	39.42	81.03	64.64	26.87	56.67
FedAVE$\|r^-$	97.24	93.43	92.89	67.08	82.69
FedAVE$\|q^-$	92.21	97.65	64.87	62.92	85.53
FedAVE	**97.59**	**97.79**	**93.62**	**84.22**	**94.78**

Ablation study. To examine the effectiveness of the two individual modules in the FedAVE, a series of ablation experiments are conducted on MNIST, as shown in Table 4. FedAVE$|r^-q^-$ denotes the model without the data distribution information. FedAVE$|r^-$ overlooks the information of the data distribution in the ARC module. FedAVE$|q^-$ lacks of the information on the data distribution in the GED module. Table 4 shows that our proposed method improves fairness compared to other variants in each scenario. For example, for the DIR (0.2) data partition on MNIST, our method improves the fairness by 57.35%, 17.14%, and 21.3%, respectively. The ablation study shows that the two designed modules in the FedAVE are indispensable and important in improving collaborative fairness.

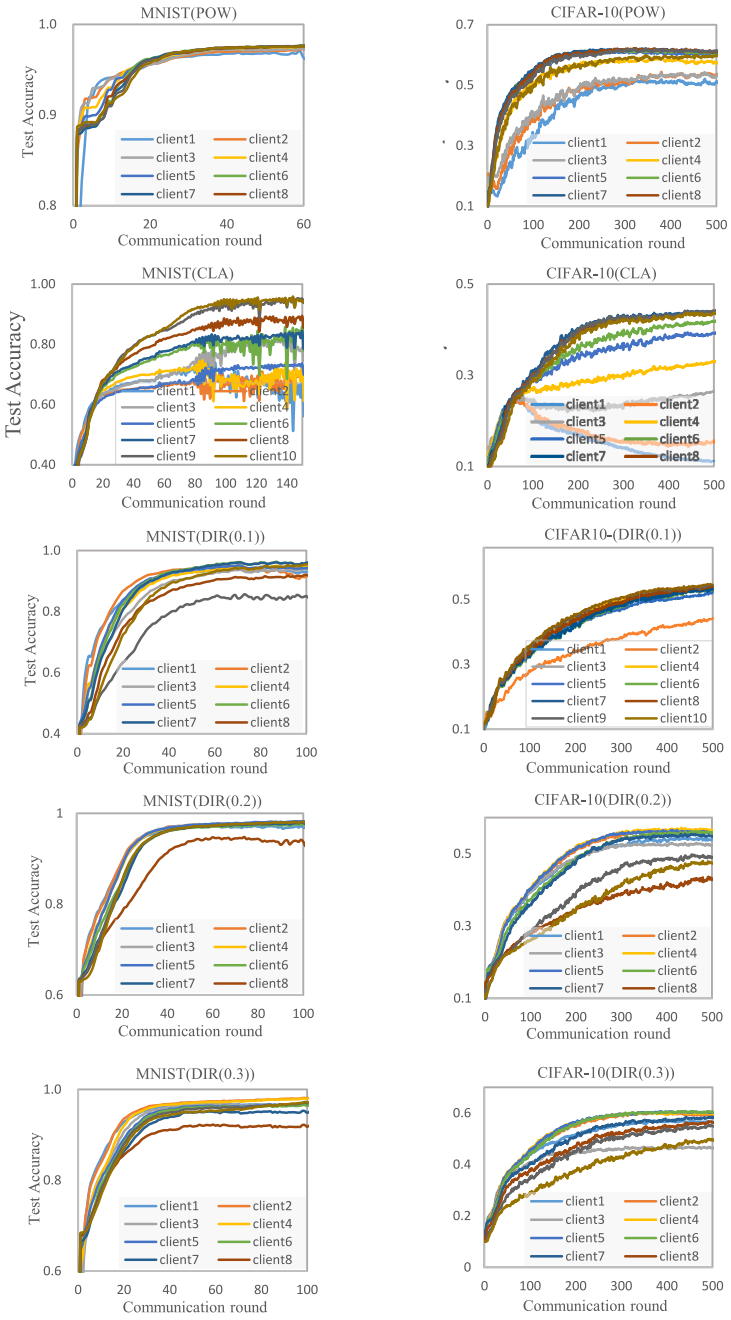

Fig. 3. Test accuracy achieved by clients for MNIST (left) and CIFAR-10 (right) in each round. From top to bottom {POW, CLA, DIR (0.1), DIR (0.2), DIR (0.3)}.

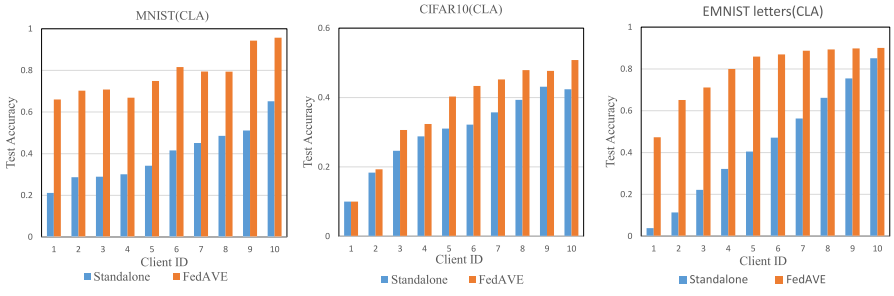

Fig. 4. Test accuracy results of Standalone and FedAVE in each client.

6. Conclusions and future work

In this work, we proposed FedAVE to fairly evaluate the reputations of clients uploading models in FL. Then, the obtained reputations are employed to design corresponding rewards in the form of gradients. Our approach ensures that the clients that contribute more achieve higher-quality gradients, resulting in better models for high-quality clients. The experiments demonstrate that the FedAVE achieves high collaborative fairness and ensures the predictive performance of each client's model in different heterogeneous settings.

Currently, the FedAVE does not consider dynamic participation in FL where clients can join or leave the system at any time. New clients who join late always have lower reputations, which means that low contributors will be judged as high contributors at this time. Hence, future work will investigate the dynamic participation in FL.

CRediT authorship contribution statement

Zihui Wang: Data curation, Formal analysis, Investigation, Methodology, Writing – original draft. **Zhaopeng Peng:** Data curation, Software, Writing – review & editing. **Xiaoliang Fan:** Conceptualization, Funding acquisition, Investigation, Methodology, Supervision, Writing – original draft, Writing – review & editing. **Zheng Wang:** Methodology, Validation. **Shangbin Wu:** Software, Visualization. **Rongshan Yu:** Supervision, Writing – review & editing. **Peizhen Yang:** Validation, Visualization. **Chuanpan Zheng:** Investigation, Methodology. **Cheng Wang:** Investigation, Resources.

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

The authors do not have permission to share data.

Acknowledgments

The research was supported by Natural Science Foundation of China (62272403, 61872306), and FuXiaQuan National Independent Innovation Demonstration Zone Collaborative Innovation Platform (No. 3502ZCQXT2021003).

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] 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.
[3] Q. Yang, Y. Liu, T. Chen, Y. Tong, Federated machine learning: Concept and applications, ACM Trans. Intell. Syst. Technol. 10 (2) (2019) 1–19.
[4] J. Xu, B.S. Glicksberg, C. Su, P. Walker, J. Bian, F. Wang, Federated learning for healthcare informatics, J. Healthc. Inform. Res. 5 (1) (2021) 1–19.
[5] N. Rieke, J. Hancox, W. Li, F. Milletari, H.R. Roth, S. Albarqouni, S. Bakas, M.N. Galtier, B.A. Landman, K. Maier-Hein, et al., The future of digital health with federated learning, NPJ Digit. Med. 3 (1) (2020) 1–7.
[6] J.M. Drazen, S. Morrissey, D. Malina, M.B. Hamel, E.W. Campion, The importance—and the complexities—of data sharing, N. Engl. J. Med. 375 (12) (2016) 1182–1183.
[7] C. Palihawadana, N. Wiratunga, A. Wijekoon, H. Kalutarage, FedSim: Similarity guided model aggregation for federated learning, Neurocomputing 483 (2022) 432–445.
[8] J. Huang, R. Talbi, Z. Zhao, S. Boucchenak, L.Y. Chen, S. Roos, An exploratory analysis on users' contributions in federated learning, in: 2020 Second IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA), IEEE, 2020, pp. 20–29.
[9] R.H.L. Sim, Y. Zhang, M.C. Chan, B.K.H. Low, Collaborative machine learning with incentive-aware model rewards, in: International Conference on Machine Learning, PMLR, 2020, pp. 8927–8936.
[10] G. Wang, C.X. Dang, Z. Zhou, Measure contribution of participants in federated learning, in: 2019 IEEE International Conference on Big Data, IEEE, 2019, pp. 2597–2604.
[11] Y. Shi, H. Yu, C. Leung, Towards fairness-aware federated learning, 2021, arXiv preprint arXiv:2111.01872.
[12] L. Lyu, J. Yu, K. Nandakumar, Y. Li, X. Ma, J. Jin, H. Yu, K.S. Ng, Towards fair and privacy-preserving federated deep models, IEEE Trans. Parallel Distrib. Syst. 31 (11) (2020) 2524–2541.
[13] X. Xu, L. Lyu, X. Ma, C. Miao, C.S. Foo, B.K.H. Low, Gradient driven rewards to guarantee fairness in collaborative machine learning, Adv. Neural Inf. Process. Syst. 34 (2021).
[14] L. Lyu, X. Xu, Q. Wang, H. Yu, Collaborative fairness in federated learning, Fed. Learn.: Priv. Incent. (2020) 189–204.
[15] J.-H. Luo, J. Wu, W. Lin, Thinet: A filter level pruning method for deep neural network compression, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 5058–5066.
[16] R. Yu, A. Li, C.-F. Chen, J.-H. Lai, V.I. Morariu, X. Han, M. Gao, C.-Y. Lin, L.S. Davis, Nisp: Pruning networks using neuron importance score propagation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9194–9203.
[17] P. Molchanov, A. Mallya, S. Tyree, I. Frosio, J. Kautz, Importance estimation for neural network pruning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 11264–11272.
[18] L. Lyu, Y. Li, K. Nandakumar, J. Yu, X. Ma, How to democratise and protect AI: Fair and differentially private decentralised deep learning, IEEE Trans. Dependable Secure Comput. 1–1 (2020).
[19] X. Xu, L. Lyu, A reputation mechanism is all you need: Collaborative fairness and adversarial robustness in federated learning, 2020, arXiv preprint arXiv:2011.10464.
[20] H. Yu, Z. Liu, Y. Liu, T. Chen, M. Cong, X. Weng, D. Niyato, Q. Yang, A fairness-aware incentive scheme for federated learning, in: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, 2020, pp. 393–399.
[21] J. Zhang, Y. Wu, R. Pan, Incentive mechanism for horizontal federated learning based on reputation and reverse auction, in: Proceedings of the Web Conference 2021, 2021, pp. 947–956.
[22] Z. Chen, Z. Liu, K.L. Ng, H. Yu, Y. Liu, Q. Yang, A gamified research tool for incentive mechanism design in federated learning, in: Federated Learning, Springer, 2020, pp. 168–175.
[23] M. Cong, H. Yu, X. Weng, S.M. Yiu, A game-theoretic framework for incentive mechanism design in federated learning, in: Federated Learning, Springer, 2020, pp. 205–222.
[24] A. Agarwal, M. Dahleh, T. Sarkar, A marketplace for data: An algorithmic solution, in: Proceedings of the 2019 ACM Conference on Economics and Computation, 2019, pp. 701–726.
[25] Y. Zhan, J. Zhang, Z. Hong, L. Wu, P. Li, S. Guo, A survey of incentive mechanism design for federated learning, IEEE Trans. Emerg. Top. Comput. (2021) 1.
[26] M. Mohri, G. Sivek, A.T. Suresh, Agnostic federated learning, in: International Conference on Machine Learning, PMLR, 2019, pp. 4615–4625.
[27] T. Li, M. Sanjabi, A. Beirami, V. Smith, Fair resource allocation in federated learning, in: International Conference on Learning Representations, 2019, pp. 1–13.
[28] Z. Wang, X. Fan, J. Qi, C. Wen, C. Wang, R. Yu, Federated learning with fair averaging, in: International Joint Conference on Artificial Intelligence, 2021, pp. 1615–1623.
[29] T. Song, Y. Tong, S. Wei, Profit allocation for federated learning, in: 2019 IEEE International Conference on Big Data, IEEE, 2019, pp. 2577–2586.
[30] Y. Lin, S. Han, H. Mao, Y. Wang, B. Dally, Deep gradient compression: Reducing the communication bandwidth for distributed training, in: International Conference on Learning Representations, 2018, pp. 1–14.
[31] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in: International Conference on Machine Learning, PMLR, 2013, pp. 1310–1318.
[32] C. Wu, F. Wu, L. Lyu, Y. Huang, X. Xie, Communication-efficient federated learning via knowledge distillation, Nat. Commun. 13 (1) (2022) 1–8.
[33] M. Chen, B. Mao, T. Ma, Fedsa: A staleness-aware asynchronous federated learning algorithm with non-IID data, Future Gener. Comput. Syst. 120 (2021) 1–12.
[34] X. Li, K. Huang, W. Yang, S. Wang, Z. Zhang, On the convergence of FedAvg on non-IID data, in: International Conference on Learning Representations, 2020, pp. 1–26.
[35] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, C. Renggli, The convergence of sparsified gradient methods, Adv. Neural Inf. Process. Syst. 31 (2018).
[36] Z. Yan, D. Xiao, M. Chen, J. Zhou, W. Wu, Dual-way gradient sparsification for asynchronous distributed deep learning, in: 49th International Conference on Parallel Processing-ICPP, 2020, pp. 1–10.
[37] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[38] A. Krizhevsky, Learning Multiple Layers of Features from Tiny Images Master's thesis, University of Tront, 2009.
[39] 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.

[40] C. Chen, J. Zhang, L. Lyu, GEAR: A margin-based federated adversarial training approach, in: Proceedings of the AAAI Workshop for Trustable, Verifiable and Auditable Federated Learning in Conjunction, 2022.
[41] Q. Li, Y. Diao, Q. Chen, B. He, Federated learning on non-iid data silos: An experimental study, 2021, arXiv preprint arXiv:2102.02079.
[42] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, N. Hoang, Y. Khazaeni, Bayesian nonparametric federated learning of neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 7252–7261.
[43] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, L. Jackel, Handwritten digit recognition with a back-propagation network, Adv. Neural Inf. Process. Syst. 2 (1989).
[44] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).
[45] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

Zihui Wang received the B.S. degree in applied chemistry from Qinghai University, Xining, China, in 2016, and the M.S. degree in chemical engineering from Xiamen University, Xiamen, China, in 2019, respectively. He is currently working toward the Ph.D. degree in computer science and technology at Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, China. His current research interests include federated learning, machine learning and graph representation learning.

Zhaopeng Peng received the B.Sc. degree in Digital media technology from Shandong University, Weihai, China, in 2022. He is currently working toward the MA.Eng. degree in computer science and technology at Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, China. His research interests include spatio-temporal data representation learning and federated learning.

Xiaoliang Fan is a Senior Research Specialist with Fujian Key Laboratory of Sensing and Computing for Smart Cites, and Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, School of Informatics, Xiamen University, China. He received his Ph.D. degree at University Pierre and Marie CURIE, France in 2012. His research interests include trustworthy AI and federated learning, spatio-temporal data mining, and services computing, etc. He has published 70+ journals (IEEE TSC/TMC/TITS, etc.) and top conferences (AAAI, IJCAI, WWW, etc.) papers. His works are funded by NSFC and many industry collaborators. Dr. FAN is an IEEE Senior Member, and CCF Senior Member.

Zheng Wang received his B.E. degree in computer science of Xiamen University and currently is a Ph.D. student in the ASC lab of Xiamen University. His research interest lies in federated learning and fairness in AI. He built a federated learning platform easyFL and contributed to the project FedSTGraph during the lab experience.

Shangbin Wu received the B.E degree in computer science and technology from the School of Information Engineering, Chang'An University, Xian, China, in 2018. He is currently working toward the M.S. degree at the Department of Computer Science, Fujian Key Laboratory of Sensing and Computing for Smart Cities and School of Informatics, Xiamen University. Xiamen, China. His current research interests include Spatio-temporal Data Mining, AI algorithms and Urban Computing.

Rongshan Yu received his bachelor degree in Electrical Engineering with minor in Applied Mathematics from Shanghai Jiaotong University, China in 1995, and the Ph.D. degree from the National University of Singapore (NUS, Singapore) in 2004. He is currently with the Department of Computer Science, Xiamen University as a professor, and vice-director of the National Institute for Data Science in Health and Medicine, Xiamen University. His research interests include high-throughput multi-omics data analysis, precision medicine, and medical artificial intelligence. He has more than 100 journal/conference publications and holds more than 20 US/International patents.

Peizhen Yang received the B.S. degree in computer science and technology from Southeast University, China, in 2020. She is a Master Student in computer science and technology at Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, China. Her research interests include federated learning, spatial–temporal data mining and graph representation learning.

Chuanpan Zheng received the B.Sc. degree in applied physics from Shandong University, Jinan, China, in 2012. He is currently working toward the Ph.D. degree in computer science and technology at Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics, Xiamen University, China. His research interests include spatio-temporal data representation learning and graph neural networks.

Cheng Wang (M'04-SM'16) received the Ph.D. degree in information and communication engineering from National University of Defense Technology, Changsha, China, in 2002. He is currently a Professor and an Associate Dean of the School of Informatics, and Director of Fujian Key Laboratory of Sensing and Computing for Smart Cities, both at Xiamen University, China. His research interests include remote sensing image processing, mobile LiDAR data analysis, and multi-sensor fusion. He has co-authored over 150 papers in referred journals and top conferences including IEEE-TGRS, PR, IEEE-TITS, AAAI, CVPR, IJCAI, and ISPRS-JPRS.