FedALP: An Adaptive Layer-Based Approach for Improved Personalized Federated Learning

Zaipeng Xie[1,2](✉), Yao Liu[2], Zhihao Qu[2], Bin Tang[2], and Weiyi Zhao[3]

[1] State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering, Hohai University, Nanjing, China

[2] College of Computer and Information, Hohai University, Nanjing, China
{zaipengxie,lyao,quzhihao,cstb}@hhu.edu.cn

[3] The University of Hong Kong, Hong Kong, China
wyzhao99@connect.hku.hk

Abstract. Personalized federated learning (PFL) is an improved framework that can facilitate the handling of data heterogeneity by learning personalized models. As personalization performance directly depends on the global model, it is desired to acquire a global model with a decent generalization capability under data heterogeneity. This paper proposes a novel PFL scheme, FedALP, integrating the clustering method with an adaptive layer-based fusion algorithm. Experiments are performed using various neural network models on three standard datasets. Experimental results demonstrate that, compared with the FedAvg method, our scheme can significantly improve the local model's performance with a negligible decrease in the generalization capability of the global model. Furthermore, our scheme is customizable for specific PFL applications; hence it may provide a flexible strategy to effectuate a balanced performance for both the global and the local models.

Keywords: Personalized federated learning · Adaptive · Layer-based · Non-IID

1 Introduction

Federated Learning (FL) is a distributed deep learning framework [11] that allows multiple clients to jointly train a shared global model under the coordination of a central server while keeping the participants' data private. Most of the existing training methods are variants of the Federated Averaging (FedAvg) introduced by McMahan et al. [14]. However, in the presence of statistical data

Supported by The Belt and Road Special Foundation of the State Key Laboratory of Hydrology-Water Resources and Hydraulic Engineering under Grant 2021490811, the National Natural Science Foundation of China under Grant No. 61872171, and the Fundamental Research Funds for the Central Universities No. B210201053.

L. Wang et al. (Eds.): WASA 2022, LNCS 13472, pp. 80–92, 2022.
https://doi.org/10.1007/978-3-031-19214-2_7

heterogeneity [20], such as non-IID and imbalanced data, it is difficult for FL to train a single model that works well for all clients. Optimizing the global model independently may result in poor performance in the local models [8,20].

Personalized federated learning (PFL) [16] has been proposed as a solution to mitigate the aforementioned issues. Many efforts [13] have been made to explore a scheme that exhibits sound global generalization properties and well personalized local matching properties. Wu et al. [17] proposed a tailored hierarchical communication architecture that introduced an intermediate layer of servers between the cloud and the clients for asynchronous training. Arivazhagan et al. [1] proposed a neural network architecture where the base layer is trained on a centralized server using FedAvg, and the top layer is trained locally using a gradient descent variant. Liang et al. [9] proposed a new FL algorithm that learns a compact local representation and a global model across all clients. However, these personalization methods are usually focused on enhancing local representations, and the generalization capability of the global model is of less concern.

In the PFL process, many clients may share some similarities in the data distribution. If these clients can be aggregated for mutual benefits, the performance may outperform localized adaptation schemes [10]. Briggs et al. [2] proposed a hierarchical clustering strategy to separate client clusters by comparing their local updates with the global model. Ma et al. [12] proposed a personalized FL method that incorporates attention-based clustering to facilitate collaborations among similar clients. Zhang et al. [19] proposed a PFL framework that can calculate optimal weighted model combinations for each client. Huang et al. [5] proposed a attentive message passing mechanism that can assist the collaboration among clients significantly. Instead of maintaining a single global model, this mechanism retains a personalized cloud model for individual client.

Sattler et al. [15] proposed clustered federated learning paradigm that exploits geometric properties of the FL loss surface to group the clients into clusters. However, these approaches do not consider the relationship between the global model and the personalized local models. The generalization performance of the global model can be impaired because of the limited communication between client groups. Wu et al. [17] pointed out that the comprehensive knowledge from the global model may be beneficial in situations when limited local data is acquired for training. On the other hand, the global model with a decent generalization performance can serve as an unbiased initialization for new users. Hence, it is desirable to explore a novel PFL training framework that ensures adequate performance for the global model.

This paper proposes a hierarchical PFL framework FedALP named *federated learning with adaptive layer-based personalization*. The focus of this framework is on addressing the above issues. Our contributions can be summarized as follows:

– We propose a novel federated learning framework, which integrates a client clustering method and an adaptive layer-based fusion algorithm. This framework does not require manual efforts, and it can adaptively allocate layers for

the personalization model to maintain a decent performance for both local and global models.
- Our proposed scheme is fully customizable for specific PFL applications; hence it can provide a flexible strategy to effectuate a balanced performance for both the global and the local models.
- Experiments have been performed on models with datasets including MNIST, FashionMNIST, and CIFAR-10. The results demonstrate that FedALP can improve the performance of the local model by maximum 31.5% with at most 6.8% decrease in the performance of the global model. FedALP on non-IID data can achieve a comparable or even better performance than the FedAvg framework on IID data for the same dataset. And as β varies, FedALP can provide a dynamic performance between an optimized global and personalized local performance.

2 Methodology

2.1 Motivation

While FL has been shown to be effective in training a single accurate global inference model, it may not generate a satisfactory global model shared by all nodes on non-IID dataset. In recent years, in the explorations of PFL, many researchers have focused on two possible solutions:

1) **Clustering-based personalization** [2,15,19]. Instead of expecting the global model to perform well on all clients, this method trains dedicated models for sharing within a group of clients with similar data distribution.
2) **Layer-based personalization** [1,3,9]. These method personalizes some layers of the local model, while the rest are derived from the global model.

However, current clustering-based personalization approaches rarely focus on model sharing between groups. Consequently, they may compromise the generalization performance of the global model. Meanwhile, current layer-based personalization approaches lack flexibility and adaptability because they usually adopt predefined layering. Therefore, they may end up with a suboptimal solution, leading to an unbalanced performance for both the global and the local models.

The proposed scheme, FedALP, employs an adaptive layer-based PFL scheme that incorporates a clustering method. In this method, the layer-based personalization scheme is applied to a group of clients. Each client can return the performance feedback within the group to regulate the layer-based personalization training.

2.2 Algorithm Design

Algorithm 1 describes the proposed scheme of FedALP, and a summary of the symbols is listed in Table 1. In general, FedALP's workflow consists of three phases. The first phase is the warm-up phase, where the FedALP initiate the global model on the global server and push it to every participating client. The training process at this phase follows the FedAvg [14] scheme and runs for T_{pre} rounds. The second phase performs the clustering and the layer-wise personalization based on the results from the warm-up phase. The third phase is the main body of FedALP when the groups and the layers are set. In this phase, the hierarchical PFL training is carried out for $(T - T_{pre})$ rounds until the models achieve satisfactory performance results. Details of the process of FedALP are elaborated as follows:

Table 1. Notation

Symbol	Explanation	Symbol	Explanation
T	Iteration number of overall training	$w_g^{(t)}$	Global model parameters of iteration t
T_{pre}	Iteration number of the warm-up phase	$w_m^{(t)}$	The m^{th} group's model parameters of iteration t
K	Number of clients	$\mathcal{G}_m$	Set of client index in m^{th} group
M	Number of groups	$\mathcal{D}_m$	Average gradient of clients within m^{th} group
ρ	Cosine similarity matrix, $\rho \subset \mathbb{R}^{K \times K}$	Ψ_m	Personalization weight of m^{th} group
l	Number of model layers involved in training	α	Dirichlet distribution parameters, $\alpha \in [0, +\infty)$
γ_k	Samples number weight of client k	β	Personalization factor, $\beta \in [0, 1]$

Warm-up Phase: At the beginning of FedALP, each client will initialize a model that is trained and shared at a given frequency (every 20 epochs in our setting). Meanwhile, the Global server receives clients' gradients to update the global model; then it pushes the latest global model to the clients. The process at the current phase is the same as the standard FedAvg's setting; the model training at iteration $t + 1$ will only begin after successfully receiving $w^{(t)}$. We train the global model for T_{pre} iterations, where T_{pre} is a predefined setting which is typically set to be 40% to 70% of the overall training iterations T. At this phase, the global objective function of FedAvg is given by

$$\min_{w} \left\{ f(w) \triangleq \sum_{k=1}^{K} \gamma_k F_k(w) \right\}, \tag{1}$$

where K is the number of clients, γ_k is the weight of the k-th client, $\gamma_k \geq 0$, $\sum_k \gamma_k = 1$, and $F_k(w)$ is the local objective functions. The local objective functions is given by

$$F_k(w) \triangleq \mathbb{E}_{(x,y)\sim p_{data}^k} L(x, y; w), \tag{2}$$

where $p_{data}^{(k)}$ is the data distribution of client k, $L(\cdot)$ is the loss function of the predictions on examples (x, y) made with model parameters w. A global model, $w_g^{(T_{pre})}$, can be obtained after T_{pre} rounds and is shared by each client.

Layer-wise Personalization with Clustering: This phase starts with a global model $w_g^{(T_{pre})}$ where its gradient updates $\{\Delta w_k^{(T_{pre})}\}_{k=1}^K$ are noted as ΔW. Algorithm 2 describes the process of this phase. A pairwise cosine similarity matrix $\rho \subset \mathbb{R}^{K\times K}$ is constructed with cosine similarity kernel S as follows:

$$\rho = S(\Delta W),\ \rho_{ij} = S_C(i,j), \tag{3}$$

where the cosine similarity $S_C(\cdot,\cdot)$ between the gradient updates of any two clients i and j is defined by:

$$S_C(i,j) \triangleq \frac{< \Delta w_i^{(T_{pre})}, \Delta w_j^{(T_{pre})} >}{||\Delta w_i^{(T_{pre})}||\,||\Delta w_j^{(T_{pre})}||}, \tag{4}$$

where $i,j \in \{1,2,\cdots,K\}$. Then we use a top-down hierarchical clustering algorithm [4] to cluster K clients into M groups based on ρ, and thus produce a group list, denoted by $\{\mathcal{G}_m\}_{m=1}^M$. A single process is designated as the group server for coordinating among clients within the group. The group server maintains a group model w_m, while the global model is denoted as w_g.

Algorithm 1: FL with Adaptive Layer-based Personalization (FedALP)

Procedure FedALP SERVER TRAINING:

Input: Round number T_{pre}, T, local epochs E, batch size B, learning rate η

Output: Global model $w_g^{(T)}$ and group models $\{w_m^{(T)}\}_{m=1}^M$

Get $w_g^{(T_{pre})}$ by **FedAvg** [14] with E, B, η, T_{pre}
Execute **FEDALP INITIALIZATION** (Algorithm 2)
Initialize group model $\{w_m^{(T_{pre})}\}_{m=1}^M$ with $w_m^{(T_{pre})} \leftarrow w_g^{(T_{pre})}$
for *each global round* $t = T_{pre+1}, T_{pre+2}, \cdots, T$ **do**
    for $m = 1,2,\cdots,M$ **do**
        $\mathcal{W}_{new}$=**MixByLayer**$(w_m^{(t-1)}, w_g^{(t)}, \Psi_m)$
        Server broadcasts $\mathcal{W}_{new}$ to client $k \in \mathcal{G}_m$
        for *each client* $k \in \mathcal{G}_m$ **do**
            $\Delta w_k^{(t+1)} \leftarrow$**ClientUpdate**$(k, \mathcal{W}_{new})$
        $w_m^{(t+1)} \leftarrow w_m^{(t)} + \sum_{k\in\mathcal{G}_m} \gamma_k \Delta w_k^{(t+1)}$
    $w_g^{(t+1)} \leftarrow \sum_{m=1}^M \left\{ (\sum_{k\in\mathcal{G}_m} \gamma_k) w_m^{(t+1)} \right\}$

Function MixByLayer(w_m, w_g, Ψ_m):
for *each model layer* $l = 1,2,\cdots,L$ **do**
    $w^{(l)} \leftarrow \Psi_m^{(l)} w_m^{(l)} + (1-\Psi_m^{(l)}) w_g^{(l)}$
return $w \leftarrow \{w^{(l)}\}_{l=1}^L$

Function ClientUpdate(i, w):
$\hat{w} \leftarrow w$
for *each local epoch* $e = 1,2,\cdots,E$ **do**
    $w \leftarrow w - \eta \cdot \nabla L(b; w)$ for local batch $b \in B_i$
return $\Delta w \leftarrow w - \hat{w}$

Next, the group server adopts the proposed adpative layer-based fusion algorithm to generate a layer-wise weight list Ψ. The procedure of getting Ψ is given as follows: Given a group $\mathcal{G}_m$, we first calculate the average gradient updates $\mathcal{D}_m$ within each group as given by:

$$\mathcal{D}_m = \sum_{k \in \mathcal{G}_m} \gamma_k \Delta w^{(k)}. \tag{5}$$

Then the updates can be divided into individual sets of layers as given by

$$\mathcal{D}_m = \left[\mathcal{D}_m^{(1)}, \mathcal{D}_m^{(2)}, \cdots, \mathcal{D}_m^{(l)}\right], \tag{6}$$

where l represents the total number of model layers involved in the training.

We define a tensor δ_m to represent the Euclidean distance of each layer in the model within m^{th} group. δ_m can be derived from $\mathcal{D}_m$ as given by

$$\delta_m \leftarrow \left\{||\mathcal{D}_m^{(1)}||_2, ||\mathcal{D}_m^{(2)}||_2, \cdots, ||\mathcal{D}_m^{(l)}||_2\right\}, \tag{7}$$

where the Euclidean norm, $||\mathcal{D}_m^{(n)}||_2$, represents the update distance of the n^{th} layer and $n \in \{1, \cdots, l\}$. It is worth noting that $||\mathcal{D}_m^{(n)}||_2$ is proportional to the degree of the personalization for the layer.

We define a personalization factor β and then the layer-based personalization weights Ψ_m is calculated as given by

$$\Psi_m = \beta \cdot \delta_m / \mathbf{max}(\delta_m). \tag{8}$$

The personalization factor β is a parameter that can have an impact on the personalization degree of FedALP. When $\beta = 0$, FedALP turns into FedAvg; when $\beta = 1$, some layers are completely localized at the expense of the generalization capability of the global model.

FedALP Hierarchical PFL Training: In this phase, the group server takes over the global sever as the organizer within each group, where clients' gradients are sent to update the group model w_m, and the latest group model is sent back to the clients. While the global server only communicates with the group servers. The global model w_g is updated by averaging the w_m at every global iteration. Figure 1 describes current phase of FedALP.

At the beginning of each iteration, the global server sends the latest global model w_g to each group server. Then, the global model is weighted and fused with the group model w_m layer-by-layer. The model parameter of the n^{th} layer at group m is given by

$$\mathcal{W}_{new}^{(n)} = \Psi_m^{(n)} w_m^{(n)} + (1 - \Psi_m^{(n)}) w_g^{(n)}, \tag{9}$$

process.pdf

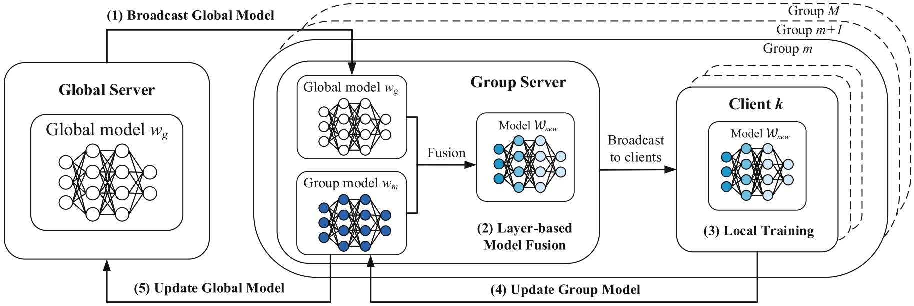

Fig. 1. FedALP training process

where $\mathcal{W}_{new}^{(n)}$ represents the n-th layer of $\mathcal{W}_{new}$ and $n \in \{1, \cdots, l\}$. For each group, the model $\mathcal{W}_{new}$ has l layers and is given by

$$\mathcal{W}_{new} \leftarrow \left\{ \mathcal{W}_{new}^{(1)}, \mathcal{W}_{new}^{(2)}, \cdots, \mathcal{W}_{new}^{(l)} \right\}. \tag{10}$$

Here $\mathcal{W}_{new}$ serves as the starting point for the next iteration and is broadcasted within the group. The client trains the model $\mathcal{W}_{new}$ for several epochs (we pick 20 as our setting) and they are aggregated to update the group model w_m.

The training process repeats until the desired number of iterations or the accuracy reaches a given threshold. Thus it concludes the FedALP process. In our method, all model aggregations are weighted based on the amount of data owned by the client to optimize the model performance further.

Algorithm 2: Layer-wise Personalization Algorithm with Clustering

Procedure LAYER-WISE PERSONALIZATION WITH CLUSTERING
  Input: Group number M, personalization factor β, gradients $\{\Delta w_k^{(T_{pre})}\}_{k=1}^K$
  Output: $\{\mathcal{G}_m\}_{m=1}^M$ and $\{\Psi_m\}_{m=1}^M$
  Estimated hierarchical clustering P with Ward method from the similarity matrix ρ, where $\rho_{i,j} = S_C(\Delta w_i^{(T_{pre})}, \Delta w_j^{(T_{pre})}), i, j \in \{k\}_{k=1}^K$ (Eq. 4)
  Intersect P to determine M groups $\{\mathcal{G}_m\}_{m=1}^M$, $\mathcal{G}_m = \{k \mid \textit{client } k \textit{ in group } m\}$
  for $m = 1, 2, \cdots, M$ **do**
    $\mathcal{D}_m \leftarrow \sum_{k \in \mathcal{G}_m} \gamma_k \Delta w_k^{(T_{pre})}$
    $\Psi_m \leftarrow$ **LayersWeight**$(\mathcal{D}_m, \beta)$

Function LayersWeight$(\mathcal{D}, \beta)$:
  $\mathcal{D} = [\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \cdots, \mathcal{D}^{(L)}]$
  for *each model layer* $l = 1, 2, \cdots, L$ **do**
    $\delta^{(l)} \leftarrow ||\mathcal{D}^{(l)}||_2$
  $\delta \leftarrow \{\delta^{(1)}, \delta^{(2)}, \cdots, \delta^{(L)}\}$
  return $\Psi \leftarrow \beta \cdot \delta / \mathbf{max}(\delta)$

In summary, the proposed method can adaptively get the optimized layer-based personalization for various models. Compared to personalizing the entire model, our layer-based personalization can improve the performance of the local model with a minimal impact on the global model generalization performance. While implementing personalization, we also optimize the global model. Hence, the global model facilitates communication among groups, and every client may obtain knowledge from the global model and avoid overfitting and a locally optimal result. The global model provides a generalization capability for other applications that may exploit its ability. It is worth mentioning that, since our layer-wise algorithm is personalized, each group can have its own layer-based weights Ψ_m, which will allow using different personalization within different groups.

Our approach is flexible compared with some state-of-the-art layer-based personalization schemes [1,3,9]. This is because the proposed layer-wise algorithm is adaptable by incorporating the personalization factor, β. A balanced performance can be achieved for both the global and the local models that are tailored for specific PFL applications. For example, when $\beta = 0$, FedALP turns into FedAvg; when $\beta = 1$, FedALP becomes a variant of [1], some layers are completely localized at the expense of the generalization capability of the global model.

3 Experiments

3.1 Datasets and Model Architectures

We evaluated the performance of FedALP with four models on three non-IID datasets based on MNIST, FashionMNIST, CIFAR-10. It is worth noting that, various kinds of non-IID data partition scheme exist and our data partition is the same as in [4].

A. **MNIST** [7]. We generated a non-IID dataset consisting of 100 clients, where each client has 500 training samples and 100 test samples that consist of only one digit. Each digit is owned by 10 clients.
B. **FashionMNIST** [18]. We follow the same procedure as MNIST to create a non-IID dataset using FashionMNIST.
C. **CIFAR-10** [6]. We partition the CIFAR-10 dataset using the Dirichlet distribution, $\mathbf{DIR}(\alpha)$, to provide the corresponding cross-category partition for each client. The parameter α controls the heterogeneity of the generated dataset. When $\alpha = 0$, it means that each client gets only one category of sample, and when $\alpha \to +\infty$, it means that all categories of sample are uniformly distributed on each client. We consider 100 clients and assign datasets to clients under the IID and Dirichlet distribution with $\alpha \in \{0.001, 0.01, 0.1\}$. Clients will have unbalanced amount of samples.

The experiments utilized four models to evaluation the FedALP scheme. The first model is a fully connected network with only one hidden layer, named MNIST-NN. The second model is a CNN named MNIST-CNN. It consists of 3 convolutional layers and 2 fully connected layers. The third one is also a CNN named CIFAR10-CNN and it consists of 3 convolutional layers and 2 fully connected layers. The fourth one is an AlexNet model accustomed to the CIFAR-10 dataset, and it consists of 5 convolutional layers and 3 fully connected layers. Both MNIST-NN and MNIST-CNN were used on the MNIST and FashionMNIST datasets, while both CIFAR10-CNN and AlexNet were used on CIFAR-10.

3.2 FedALP Evaluation

In our experiments, the FedAvg algorithm [14] is used as the baseline for training on both the IID dataset and the non-IID dataset. In each experiment, the global model maintained by the FedALP algorithm is named $FedALP\_global$. Three sets of experiments have been performed.

1. Experiment 1 describes the comparison of the accuracy performance between FedAvg and FedALP on both IID and non-IID datasets.
2. Experiment 2 compares the model's accuracy performance of FedAvg and FedALP on both IID and non-IID CIFAR-10 by varying α, the degree of non-IID in datasets.
3. Experiment 3 compares the model's accuracy performance of FedALP on non-IID CIFAR-10 by varying β, the personalization factor of FedALP.

Experiment 1: Fig. 2 describes the experimental results and the accuracy values are listed in Table 2. This experiment evaluates the accuracy performance of all three datasets in four different cases: (1) FedAvg on IID datasets (green line), (2) FedAvg on non-IID datasets (orange line), (3) the **global model** performance

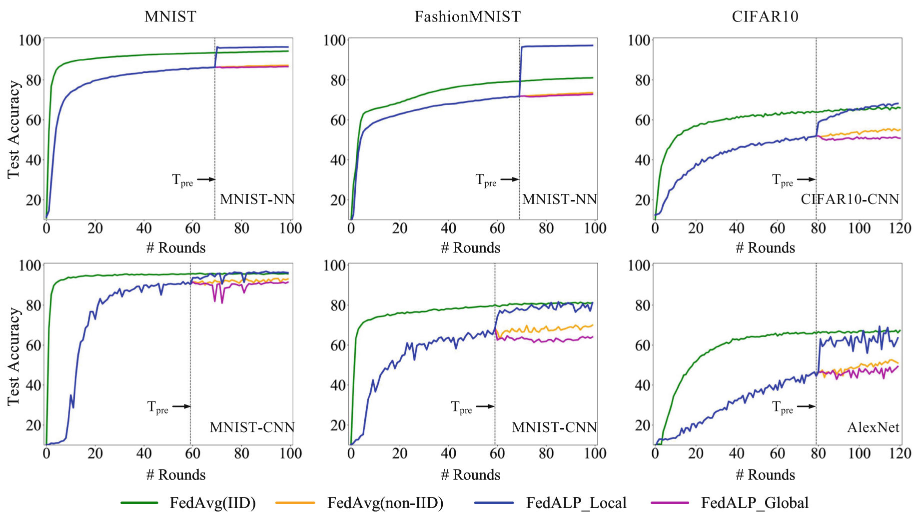

Fig. 2. Comparison of FedAvg on IID, FedAvg on non-IID, and FedALP on non-IID datasets ($\alpha = 0.001$, $\beta = 0.6$) with various models. (Color figure online)

Table 2. The accuracy comparison of FedAvg and FedALP with various models.

Dataset	Model	Accuracy				Ratio
		Non-IID ($\alpha = 0.001$)			IID	
		FedAvg	FedALP_global	**FedALP**	FedAvg	
MNIST	MNIST-NN	87.25	(↓ 0.69%)86.65	(↑ 10.56%)94.46	94.37	102.21%
	MNIST-CNN	92.55	(↓ 0.75%)91.86	(↑ 4.45%)96.67	96.04	100.66%
FashionMnist	MNIST-NN	73.48	(↓ 0.91%)72.81	(↑ 31.52%)96.64	81.05	119.24%
	MNIST-CNN	68.86	(↓ 5.63%)65.01	(↑ 15.65%)79.67	80.99	98.37%
CIFAR10	CIFAR10-CNN	54.86	(↓ 6.33%)51.39	(↑ 23.57%)67.79	65.82	102.99%
	CIFAR10-AlexNet	51.73	(↓ 6.79%)48.22	(↑ 22.83%)63.54	66.82	91.60%

of FedALP on non-IID datasets, $\alpha = 0.001$ (magenta line), (4) the average **local model** performance of FedALP on non-IID datasets, $\alpha = 0.001$ (blue line).

We observe that FedAvg on non-IID data significantly decreases its accuracy performance compared with FedAvg on IID data. Since the starting point of the FedALP is set to be at iteration T_{pre}, a notable performance enhancement is demonstrated, as shown in Fig. 2. We found that our FedALP outperforms FedAvg on non-IID by maximum 31.5% in the average local model performance, while a slight decrease (maximum 6.8%) is observed in the global model performance. The adaptively layer-based fusion method can accommodate some layers by adjusting their personalization contribution, preventing the overall model from deviating too far from the global model. Interestingly, our FedALP method on non-IID datasets and the FedAvg method on IID datasets are comparable in accuracy performance. An intuitive explanation is that since our approach can adaptively adjust the personalized layering scheme for each group, it may boost the accuracy performance even with data discrepancy.

Experiment 2: Experimental results is summarized in Fig. 3 and the accuracy values are listed in Table 3. This experiment evaluates the test accuracy performance of CIFAR-10 by setting α to three different values $\{0.001, 0.01, 0.1\}$. Results are collected for four cases: (1) FedAvg on IID datasets (green line), (2) FedAvg on non-IID datasets (orange line), (3) the **global model** performance of FedALP on non-IID datasets (magenta line), (4) the average **local model** performance of FedALP on non-IID datasets (blue line).

We observe that in all cases with non-IID datasets, our FedALP outperforms the FedAvg methods. In addition, we observe that our FedALP approach demonstrates excellent effectiveness in the accuracy performance as α decreases. This is because α is a parameter that determines the degree of non-IID, and the reduction in α will produce a performance degradation on FedAvg. At the same time, our FedALP method can mitigate the data discrepancy and boost performance.

Experiment 3: In this experiment, we evaluate the accuracy of both the global and local models by varying the personalization factor, β, as shown in Fig. 4. Two models, including CIFAR10-CNN and AlexNet, are employed on the non-IID CIFAR-10 datasets ($\alpha = 0.001$). This experiment aims to demonstrate the dynamic performance of FedALP that may be regulated for a balanced global and personalized local performance.

In Fig. 4, we compare the accuracy versus rounds by varying β from 0 to 1 with 0.3 as the step. It is worth noting that the solid lines describe the average result for the local models, and the dash-dotted lines illustrate the results for the global model.

We observe that the average local model's accuracy improves significantly as β increases-i.e., the degree of personalization of each layer increases, resulting in an improved local model. Meanwhile, the global model's accuracy degrades slightly. When $\beta = 0$, our proposed scheme produces the exact results as FedAvg. This is because all the layers contribute to the training of the global model. When $\beta = 1$, the personalization layers do not contribute to the training of the global model; hence, a maximum local model accuracy can be attained with a 12% decrease in the global accuracy performance compared with FedAvg.

In conclusion, by carefully choosing β, our scheme can significantly improve the local model's performance with a negligible decrease in the global model's accuracy. Our proposed method can adaptively accommodate specific PFL applications, providing flexibility to produce a balanced performance for both the global and the local models.

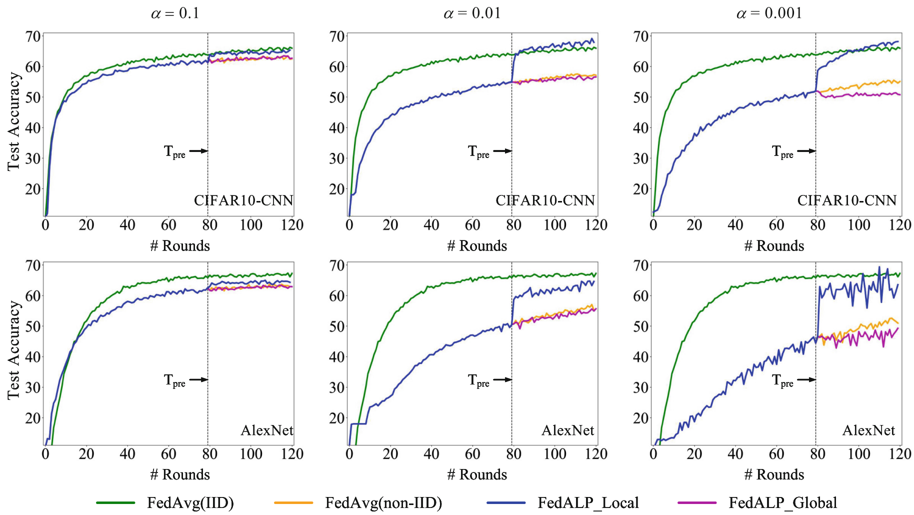

Fig. 3. Comparison of FedAvg on IID and non-IID data, FedALP on non-IID CIFAR-10 datasets with two models by varying $\alpha \in \{0.001, 0.01, 0.1\}$. (Color figure online)

Table 3. The accuracy comparison of FedAvg and FedALP by varying α.

Dataset-model	α	Accuracy		
		FedAvg	FedALP_global	FedALP
CIFAR10-CNN	0.1	62.81	(↑ 1.11%)63.51	(↑ 3.30%)64.88
	0.01	57.03	(↓ 0.79%)56.58	(↑ 19.48%)68.14
	0.001	54.86	(↓ 6.33%)51.39	(↑ 23.57%)67.79
CIFAR10-AlexNet	0.1	63.19	(↑ 0.43%)62.92	(↑ 1.88%)63.38
	0.01	56.06	(↓ 2.23%)54.81	(↑ 13.66%)63.72
	0.001	51.73	(↓ 6.79%)48.22	(↑ 22.83%)63.54

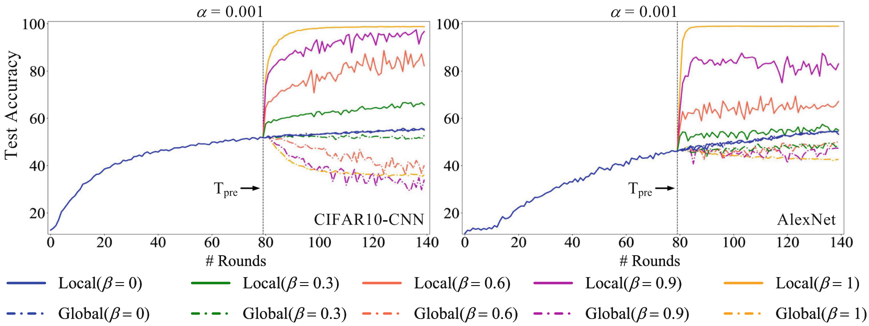

Fig. 4. FedALP on non-IID CIFAR-10 with various $\beta \in \{0, 0.3, 0.6, 0.9, 1\}$.

4 Conclusion

This study describes a novel personalization federated learning method that utilizes adaptive layer-based personalization and a clustering method. Experimental results show that the proposed method can significantly improve the local model's performance with a negligible decrease in the generalization capability of the global model. The training results on non-IID data with FedALP are comparable to a standard FedAvg on the IID data. Results also reveal that our scheme can provide a flexible strategy that effectuates a balanced performance for both the global and the local models for specific PFL applications.

References

1. Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S.: Federated learning with personalization layers. arXiv preprint arXiv:1912.00818 (2019)
2. Briggs, C., Fan, Z., Andras, P.: Federated learning with hierarchical clustering of local updates to improve training on non-IID data. In: 2020 International Joint Conference on Neural Networks (IJCNN), pp. 1–9 (2020)
3. Bui, D., et al.: Federated user representation learning. arXiv preprint arXiv:1909.12535 (2019)

4. Fraboni, Y., Vidal, R., Kameni, L., Lorenzi, M.: Clustered sampling: low-variance and improved representativity for clients selection in federated learning. In: Proceedings of the 38th International Conference on Machine Learning, ICML, 18–24 July 2021, vol. 139, pp. 3407–3416 (2021)
5. Huang, Y., et al.: Personalized cross-silo federated learning on non-IID data. In: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI. pp. 7865–7873 (2021)
6. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Technical report, University of Toronto (2009)
7. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
8. Li, X., Huang, K., Yang, W., Wang, S., Zhang, Z.: On the convergence of FedAvg on non-IID data. In: 8th International Conference on Learning Representations (ICLR). arXiv preprint arXiv:1907.02189 (2020)
9. Liang, P.P., et al.: Think locally, act globally: federated learning with local and global representations. arXiv preprint arXiv:2001.01523 (2020)
10. Liu, B., Guo, Y., Chen, X.: PFA: privacy-preserving federated adaptation for effective model personalization. In: The Web Conference 2021, pp. 923–934 (2021)
11. Lo, S.K., Lu, Q., Wang, C., Paik, H., Zhu, L.: A systematic literature review on federated machine learning: from a software engineering perspective. ACM Comput. Surv. **54**(5), 95:1-95:39 (2021)
12. Ma, Z., Lu, Y., Li, W., Yi, J., Cui, S.: PFedAtt: attention-based personalized federated learning on heterogeneous clients. In: Asian Conference on Machine Learning, ACML, vol. 157, pp. 1253–1268 (2021)
13. Mansour, Y., Mohri, M., Ro, J., Suresh, A.T.: Three approaches for personalization with applications to federated learning. arXiv preprint arXiv:2002.10619 (2020)
14. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS, vol. 54, pp. 1273–1282 (2017)
15. Sattler, F., Müller, K., Samek, W.: Clustered federated learning: model-agnostic distributed multitask optimization under privacy constraints. IEEE Trans. Neural Netw. Learn. Syst. **32**(8), 3710–3722 (2021)
16. Tan, A.Z., Yu, H., Cui, L., Yang, Q.: Towards personalized federated learning. IEEE Transactions on Neural Networks and Learning Systems, pp. 1–17 (2022)
17. Wu, R., Scaglione, A., Wai, H., Karakoç, N., Hreinsson, K., Ma, W.: Federated block coordinate descent scheme for learning global and personalized models. In: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI (2021)
18. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747 (2017)
19. Zhang, M., Sapra, K., Fidler, S., Yeung, S., Alvarez, J.M.: Personalized federated learning with first order model optimization. In: 9th International Conference on Learning Representations (ICLR) (2021)
20. Zhu, H., Xu, J., Liu, S., Jin, Y.: Federated learning on non-IID data: a survey. Neurocomputing **465**, 371–390 (2021)