Personalized Federated Learning with Feature Alignment via Knowledge Distillation

Guangfei Qi[1], Zhihao Qu[1](✉), Shen-Huan Lyu[1,2], Ninghui Jia[1], and Baoliu Ye[2]

[1] Key Laboratory of Water Big Data Technology of Ministry of Water Resources,
College of Computer Science and Software Engineering, Hohai University
[2] National Key Laboratory for Novel Software Technology, Nanjing University
{qiguangfei,quzhihao,lvsh,hhu_jnh}@hhu.edu.cn
{yebl}@nju.edu.cn

Abstract. Personalized Federated Learning (PFL) has gained significant attention for its ability to handle heterogeneous data effectively. Parameter decoupling is a typical approach to PFL. It decouples the model into a feature extractor and a classifier head, where the feature extractor is trained collaboratively to learn a common representation and the classifier head is personalized for local data. Since local training only learns personalized feature information and ignores global information, the generalization ability of the feature extractor is limited. To improve the performance of the local model, a feasible approach is to make the local feature extractor more generalized. However, prior work requires the transmission of additional feature data beyond the transmission of model parameters, which leads to privacy leakage and higher communication overhead. To address these shortcomings, we propose a PFL algorithm with feature alignment via knowledge distillation, named **PFAKD**. PFAKD enhances the training of local feature extractors by explicitly aligning each sample's local features with global features, providing more detailed guidance. Meanwhile, it avoids additional communication overhead and the risk of privacy leakage. We conduct extensive experiments in heterogeneous data scenarios. PFAKD outperforms other state-of-the-art methods by up to 4.35% in terms of model accuracy. Our code is available at https://github.com/fei0829/PFAKD.

Keywords: Personalized Federal Learning · Data Heterogeneity · Feature Alignment · Knowledge Distillation.

1 Introduction

In recent years, federated learning (FL) [8, 13, 14] has gained increasing attention due to the proliferation of mobile devices and advancements in edge computing technologies. FL aims to develop a global model across multiple devices by sharing only the model updates—gradients or parameters—without exposing users' private data. Despite its successes in enhancing data privacy and security, FL faces several challenges, with data heterogeneity being one of the most critical [9]. Data generated at the client side often leads to varied data distributions among participants, resulting in non-independent and identically distributed (non-IID) data issues. Non-IID data in FL causes "client drift" [10], where the local update direction deviates from the intended global update direction. This deviation can significantly slow down model convergence and degrade overall performance [10,17]. Consequently, in the classical FL algorithm such as FedAvg, local models may not be ideally suited for each individual client due to the variance in global and local data distributions [22].

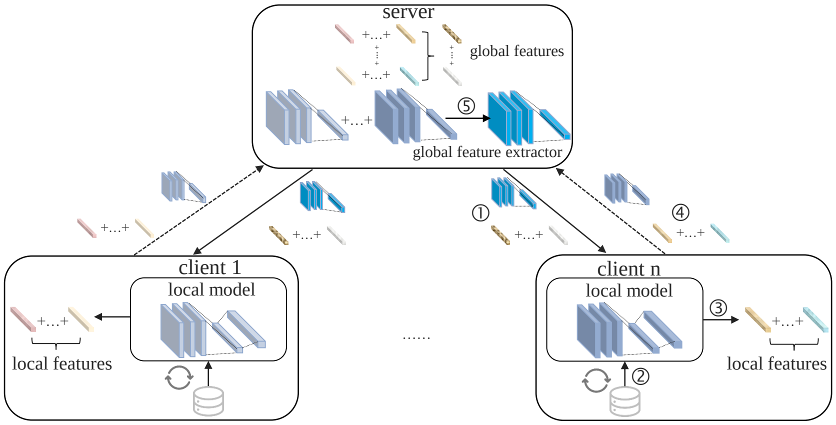


Fig. 1. The training process of existing feature alignment work: ① Download the global feature extractor and the global features of each class ② The client trains the model using local data under the guidance of global features ③ Use the locally trained feature extractor to extract local features for each class ④ Upload the local feature extractor and the local features of each class ⑤ The server aggregates the local feature extractors and local features of each class to obtain a new global feature extractor and global features of each class.

To tackle the adverse effects of data heterogeneity, personalized federated learning (PFL) [6, 15] has been proposed. PFL aims to improve local model performance by creating personalized models for each client that align with their specific data distribution. A key research direction in PFL is model decoupling [2,4,5,21], which splits the model into feature extractors and task-specific classifiers. Feature extractors are co-trained by all clients to learn a common representation, while classifiers are privately trained for local classification tasks. However, local training of feature extractors often focuses solely on personalized features, neglecting global features, which affects the aggregation effect of the global model [18, 21]. In addition, sharing feature extractors only from the parameter level is not sufficient to obtain common features from heterogeneous data.

Some recent studies have been proposed to additionally learn global features from the feature level besides the shared feature extractor. For example, Fed-

PAC [18] aligns local features with the global feature centroids, and GPFL [21] aligns them with the global category embeddings in order to introduce the global feature information into the local training. However, these approaches require additional communication of global feature data, as illustrated in Fig. 1, which introduces privacy and communication overhead issues.

To simultaneously learn personalized and global features while avoiding additional data transmission, we propose a new PFL framework with feature alignment via knowledge distillation, called **PFAKD**. In this framework, only the parameters of the feature extractor are communicated between the client and the server to facilitate the learning of a common representation. Meanwhile, the classifier head remains localized for personalized training. Clients enhance their personalized models by learning global features from the global feature extractor via distillation training, which aligns the local features of each sample with global features. PFAKD provides finer-grained feature guidance, introducing comprehensive and rich global feature information for local feature extractors, thereby reducing the diversity of feature extractors. Unlike strategies relying on global feature centroids or category embeddings, PFAKD considers the unique features of individual samples.

Our contributions are summarized as follows:

- We propose a novel PFL framework named PFAKD for feature information transfer via knowledge distillation, which allows clients to learn both personalized and global feature information. Our framework can improve the generalization ability of local feature extractors to some extent.
- PFAKD enables fine-grained feature alignment with only the parameters of the feature extractor being communicated, reducing communication overhead while maintaining model performance.
- Extensive experiments on various datasets and models demonstrate that PFAKD consistently outperforms benchmark methods, improving average model accuracy by up to 4.35%.

2 RELATED WORK

2.1 Data Heterogeneity in Federated Learning

Non-IID data is a key challenge in FL, which can significantly degrade the performance of FL models. In order to mitigate the negative impact of non-IID data on FL, related research works are divided into the following two main categories: correcting local update direction and adjustment in the model aggregation phase. Each of two strategies tries to optimize the training process from different perspectives to improve the robustness of FL with data heterogeneity.

Correcting local update direction. This type of method aims to correct the updating direction of the local model so that the local optimization objective and the global optimization objective are as consistent as possible. FedProx [12] limits the discrepancy between the global and local models by introducing a regularization term in the local training. Scaffold [10] mitigates client drift induced by non-IID by introducing two control variables. FedNova [16] reduces the variability between client gradients by normalizing and scaling the local gradients. MOON [11] exploits the idea of contrast learning by using contrast loss to correct for local training of clients. FedDyn [1] aligns the local optimization objective with the global optimization objective by introducing a dynamic regularizer for client training.

Adjustment in the model aggregation phase. This approach aims to improve the aggregation phase of the model. FedDisco [19] utilizes the difference between the client's dataset size and local-global category distributions to determine more discriminative aggregation weights for each client, and [3] proposes an elastic aggregation method that leverages the sensitivity of a parameter to adjust the update magnitude of the parameter. FedAvgM [7] improves on FedAvg by applying momentum to global model updates on the server.

2.2 Model Decoupling for Personalized Federated Learning

PFL has been proposed to address data heterogeneity in FL, with the core idea of training personalized models for each client adapted to its data distribution. One important direction is to decouple the model into a feature extractor (body) and a classifier head (head). This method trains only one of them with other clients to learn global information and the other part is used to privately learn local personalized information. FedPer [2] divides the model into a base layer and a personalized layer, with the base layer shared for training and the personalized layer trained locally and privately. FedRep [5] divides the model into feature extractors and classifier heads, where feature extractors are trained locally less often than classifier heads. FedPAC [18] uses the global feature centroids to guide the training of the local feature extractor and merge the classifier heads of clients with similar data distributions to obtain a better personalized model. GPFL [21] applies trainable global category embeddings to guide the training of the local feature extractor with the help of the Conditional Valve to learn both personalized and global information.

Although existing research has made some progress in dealing with data heterogeneity in federated learning, these works still suffer from some shortcomings. These methods often fail to fully utilize the combination of global and local feature information, thus limiting the generalization ability of feature extractors in diverse data environments. To address this problem, we propose a new personalized federated learning method, PFAKD, which achieves a more effective integration of global and local features by using the outputs of the global feature extractor to guide the training of the local feature extractor. PFAKD not only improves the generalization ability of the feature extractor, but also enhances the model's adaptability to the specific local data distribution.

3 Method

In this section, we begin with a problem statement, then introduce local representation learning and the motivation for our approach, and finally present our method PFAKD to achieve feature alignment for better personalized models through knowledge distillation.

3.1 Problem Statement

We consider a PFL system that consists of n clients, its global optimization objective is:

$$\min_{\boldsymbol{\theta}} F(\boldsymbol{\theta})\colon= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_{(x,y)\sim D_i}[\ell_i(\theta_i; x, y)] \tag{1}$$

where $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_n\}$. We denote the local model of the i-th client as θ_i, the local dataset of i-th client containing N_k data points as $\mathcal{D}_i = \{(x_{k,i}, y_{k,i})\}_{i=1}^{N_k}$ which is randomly sampled from the local data distribution P_i(x,y) in a data heterogeneous environment, and the local loss function used to measure the loss as ℓ_i. By minimizing the global loss function $F(\theta)$, we can obtain a personalized model for each client.

3.2 Local Representation Learning

We decouples the model into a feature extractor ($\boldsymbol{\phi}$) and a classifier head ($\boldsymbol{\chi}$), where $\boldsymbol{\chi}$ is the last fully-connected layer. $\boldsymbol{f_{\phi}}$ is a function parametrized by $\boldsymbol{\phi}$ that maps data points from the d-dimensional to the k-dimensional feature space $\boldsymbol{\phi}$: $\mathbb{R}^d \to \mathbb{R}^k$. $\boldsymbol{f_{\chi}}$ is a function parametrized by $\boldsymbol{\chi}$ that maps k-dimensional features to the label space $\boldsymbol{\phi} : \mathbb{R}^k \to$ y. Thus, the local loss function of a client can be expressed as: $\boldsymbol{\ell}((\boldsymbol{\phi}, \boldsymbol{\chi}) = \boldsymbol{\ell}((\boldsymbol{\phi}) \circ \boldsymbol{\ell}((\boldsymbol{\chi})$. We share the feature extractor with other clients to learn a globally common feature representation, and the classifier head is private to learn local personalized information. However, due to the effect of non-IID data, local training of clients tends to make the feature extractor overfit the local feature representation, which results in a diversity of feature extractors across clients. Therefore, the global feature extractor is not applicable to individual clients.

3.3 Motivation

Under non-IID scenarios, there are significant differences between client data distributions, and local data distributions are not representative of global data distributions. Global feature extractors are able to extract better feature representations than feature extractors trained on a skewed subset of clients. Therefore, one of our motivations is to use the global features extracted by the global feature extractor to guide the training of the local feature extractor. Knowledge distillation is a powerful knowledge transfer method that enables knowledge transfer between two models. In order to make the local feature extractor learn both global features and local personalized feature information, we propose to use knowledge distillation in order to further transfer the knowledge from the global feature extractor to the local feature extractor, which improves the generalization ability of the local feature extractor without introducing additional communication overhead.

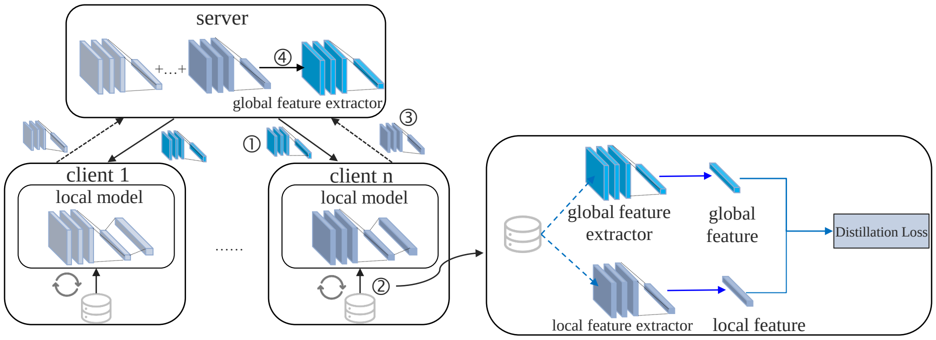


Fig. 2. The PFAKD training process: ① Download global feature extractor ② Client distillation training using local data ③ Upload local feature extractor ④ Server aggregates
local feature extractor to get global feature extractor

3.4 PFAKD

Our core goal is to transfer global feature information to local feature extractors through knowledge distillation. The features obtained from the global feature extractor are more generalized, and using them to supervise the training of the local feature extractor can be efficient in limiting the diversity of the local feature extractor parameters. Since the global features come from the global model, the client does not need to communicate additional feature information with the server, which reduces the risk of privacy leakage and further reduces communication overheads.

PFAKD performs local distillation through a linear combination of local empirical risk loss and distillation loss:

$$\boldsymbol{\ell_i(\theta_i)} = \boldsymbol{\ell_i^{ce}(\theta_i)} + \beta \cdot \boldsymbol{\ell_i^{d}(\phi_i)} \tag{2}$$

where:

$$\boldsymbol{\ell_i^{d}(\phi_i)} = \frac{1}{n_i} \sum_{m=1}^{n_i} \| \boldsymbol{f_{\phi_i}}(x_m) - \boldsymbol{f_{\phi}}(x_m) \|_2^2 \tag{3}$$

where $\boldsymbol{\ell_i^{ce}}$ is the client-side local cross-entropy loss, $\boldsymbol{\ell_i^{d}}$ is the distillation loss, here we use mean squared error MSE for distillation loss, n_i is the number of data samples, $\boldsymbol{f_{\phi_i}}(x_m)$ is the local feature obtained by the data sample x_m on client i through the local feature extraction layer $\boldsymbol{\phi_i}$, and $\boldsymbol{f_{\phi}}(x_m)$ is the global feature obtained by the data sample x_m through the global feature extraction layer $\boldsymbol{\phi}$. β is a hyperparameter balancing the local cross-entropy loss $\boldsymbol{\ell_i^{ce}}$ and distillation loss $\boldsymbol{\ell_i^{d}}$ to control the extent of knowledge transfer from the global feature extractor to the local feature extractor. By minimizing the local loss $\boldsymbol{\ell_i}$, the client can use local data to learn personalized heads, and also explicitly align local features with global features, and the local feature extractor can learn both local and global feature information to reduce the diversity of the local feature extractor

Algorithm 1 PFAKD

```
**Input:** $n$: number of clients; $\eta$: local learning rate; $\beta$: hyperparameters; $T$: global rounds; $E$: local epoch; B: number of batch
Initialize $\phi^0$, $\chi_0^0$, $\chi_1^0$,. . . , $\chi_{n-1}^0$
**for** $t = 0, 1, \ldots, T-1$ **do**
    Server samples clients $S_t$
    sends $\phi^t$ to $S_t$
    **for** each client $k \in S_t$ in parallel **do**
        Update local feature extractor: $\phi_k^{t-1} \leftarrow \phi^t$
        **for** local steps $e = 0, \ldots, E-1$ **do**
            **for** batches $j = 0, \ldots, B-1$ **do**
                $\phi_k^{(t,j+1)} \leftarrow \phi_k^{(t,j)} - \eta \nabla_{\phi_k^{(t,j)}} \ell_i$
                $\chi_k^{(t,j+1)} \leftarrow \chi_k^{(t,j)} - \eta \nabla_{\chi_k^{(t,j)}} \ell_i$
            **end for**
        **end for**
    **end for**
    Upload $\phi_k^t$ to server
    Server Aggregation: $\phi^{(t+1)} = \frac{1}{|S_t|} \sum_{i \in S_t} \phi_i^t$
**end for**
**Output:** Personalized model parameters $\{\theta_i, \ldots, \theta_n\}$
```

and promote the global aggregation of the feature extractor. We describe the algorithmic flow and training process of PFAKD in detail in Algorithm 1 and Fig. 2 respectively.

4 Experiments

4.1 Experimental Setup

Datasets and models. We evaluate our method on three commonly used image classification datasets Fashion-MNIST, CIFAR10, and CIFAR100. Fashion-MNIST is an image dataset with 10 clothing categories, CIFAR10 is an image dataset with 10 natural scene categories covering a wide range of everyday objects, and CIFAR100 is a more challenging image dataset with 100 categories covering a wider range of objects and scenes. For Fashion-MNIST and CIFAR10, we build a 5-layer CNN model with three convolutional layers and two fully connected layers, and for CIFAR100, we use a ResNet18 model.

Statistically heterogeneous settings. We use the Dirichlet distribution [20] to divide non-IID datasets, which is used to simulate the heterogeneity of data distribution in the real world. Different levels of heterogeneity can be achieved by adjusting the parameter α. The larger α is, the lower the degree of heterogeneity, and the smaller α is, the greater the degree of heterogeneity. For all methods, we used the Dirichlet distribution Dir(α) with $\alpha = 0.5$ to randomly sample data from Fashion-MNIST, CIFAR10, and CIFAR100 and divide it across clients. For all datasets, we used 75 % of the data as the training set and 25% of the data as the test set, with the training and test sets on each client having the same data distribution.

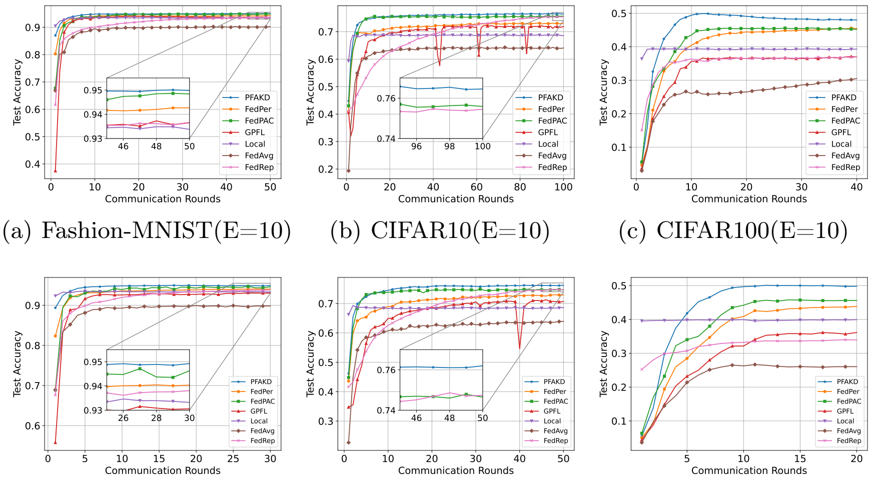
Table 1. Averaged Test Accuracy (%) of different FL methods on Fashion-MNIST, CIFAR10 and CIFAR100 with participation rate r=1.

Method	Fashion-MNIST			CIFAR10			CIFAR100		
Local epoch	5	10	20	5	10	20	5	10	20
Local	93.43			68.70			39.26		
FedAvg	90.14	90.12	89.81	64.85	64.05	62.72	34.59	32.54	26.12
FedPer	94.24	94.17	94.00	73.99	72.98	72.78	46.74	45.26	43.22
FedRep	91.90	93.48	93.57	69.00	75.37	74.30	41.04	37.13	34.31
FedPAC	94.77	94.72	94.26	75.91	75.58	74.69	47.29	45.37	45.58
GPFL	94.04	93.61	92.92	70.09	71.76	70.59	40.85	36.74	35.61
PFAKD	**94.95**	**94.99**	**94.85**	**76.12**	**76.50**	**76.15**	**48.61**	**48.10**	**49.94**

Table 2. Averaged Test Accuracy (%) of different FL methods on Fashion-MNIST, CIFAR10 and CIFAR100 with participation rate r=1.

Method	Fashion-MNIST			CIFAR10			CIFAR100		
Local epoch	5	10	20	5	10	20	5	10	20
PFAKD	94.93	94.98	94.89	76.02	76.25	76.05	47.86	48.07	49.71
PFAKD + AT	**94.97**	**94.98**	**95.02**	**76.30**	**76.74**	**76.92**	**48.27**	**48.31**	**50.10**

Baselines. We compare our proposed PFAKD with the following baselines: FedAvg [13]: train a single global model to be applied to all clients; Local: each client only uses local data to train local models, no collaborative training with other clients; FedPer [2]: decoupling the model into a base layer and a personalization layer, with the base layer shared and the personalization layer private; FedRep [5]: decoupling the model into a feature extractor and a classifier head, with the feature extraction layer shared and the classifier layer private, with the feature extractor and classifier head trained separately locally, with the feature extractor trained locally in fewer rounds compared to the classifier head; FedPAC [18]: align local features with the global feature centroids and use the global feature centroids to guide the training of the local feature extractor; GPFL [21]: by training global category embeddings for each class, the local features are aligned with the global category embeddings, and the global category embedding layer is used to guide the training of the local feature extractor.



(a) Fashion-MNIST(E=10) (b) CIFAR10(E=10) (c) CIFAR100(E=10)

(d) Fashion-MNIST(E=20) (e) CIFAR10(E=20) (f) CIFAR100(E=20)

Fig. 3. Accuracy Curves for Different Methods at Fashion-MNIST, CIFAR10, CIFAR100, and (local epochs)E=10 or E=20.

Training Settings. For all methods, we set the learning rate to 0.01, the batch size to 128, the number of clients to 10, all using the SGD optimizer with momentum set to 0.9, weight decay set to 5e-4, and local epochs set to 5, 10 and 20, respectively. For Fashion-MNIST, the number of global communication rounds is set to 50 when the local epoch is 5 or 10, and 30 when the local epoch is 20. For CIFAR10, the number of global communication rounds is set to 100 when the local epoch is 5 or 10, and 20 when the local epoch is 100. For CIFAR100, the number of global communication rounds is set to 40 when the local epoch is 5 or 10, and to 20 when the local epoch is 20. we compute the average test accuracy of the last ten communication rounds across all clients as the final test accuracy of the model.

Hyper-parameter Settings. For all methods, we use a grid search to find the optimal hyperparameters. For PFAKD, we tune β over $\{0.1, 0.5, 1, 5, 10\}$ and set to 1. For FedPAC, we tune λ over $\{0.001, 0.01, 0.1, 1, 5, 10\}$, set λ to 1 when using Fashion-MNIST and CIFAR10, and set λ to 0.001 when using CIFAR100. For GPFL, we tune λ and μ over $\{0, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$, when using Fashion-MNIST and CIFAR10, set λ and μ to 10^{-2}, 10^{-1}, respectively, and when using CIFAR100, set λ and μ to 10^{-4}, 10^{-1}, respectively.

4.2 Experimental Results

Performance Comparison. The experimental results of all methods in the case of non-IID are shown in Table **??**. In the Fashion-MNIST dataset and with local epoch of 5, 10, and 20, our method outperforms the best baseline by 0.18%, 0.27%, and 0.59%, respectively. In the CIFAR10 dataset and with local epoch of 5, 10, and 20, our method outperforms the best baseline by 0.21%, 0.92%, and 1.46%, respectively. In the CIFAR100 dataset and with local epoch of 5, 10, and 20, our method outperforms the best baseline by **1.32%**, **2.73%**, and **4.35%**, respectively.

Ablation Studies. We have two key design components in PFAKD, i.e., feature extractor sharing(FS) and knowledge distillation(KD). As demonstrated in Table 3, ("w/o" is short for "without") "w/o FS" indicates that neither FS nor KD is used, meaning only local training methods are applied. "w/o KD" refers to the training method that uses a shared feature extractor. "Both" involves feature alignment through knowledge distillation based on the shared feature extractor. Our ablation experiment results are obtained from the experimental results in Table **??**, corresponding to the three methods of Local, FedPer and PFAKD respectively. Experimental results indicate that both approaches contribute to an improvement in average test accuracy. Moreover, the combination of both methods achieves the most satisfactory model performance. This suggests that our proposed approach can build a better global feature extractor and a more suitable personalized classifier.

Table 3. Ablation study. "*w/o* FS" means local training only, "*w/o* KD" denotes feature extractor sharing , while "Both" means FS and KD are applied simultaneously.

Dataset	***w/o*** FS	***w/o*** KD	Both
Fashion-MNIST	93.43	94.17	**94.99**
CIFAR10	68.70	72.98	**76.50**
CIFAR100	39.26	45.26	**48.10**

Reason of PFAKD outperforms other baselines. (1) **PFAKD v.s. Local & FedAvg**: In scenarios where client nodes only participate in local training, they are limited to acquiring localized feature information, resulting in poor generalizability of the feature extractor. The FedAvg algorithm trains an identical model across all client nodes. However, in the presence of significant data heterogeneity, this model is often unsuitable for the local data distributions of individual clients. In contrast, the PFAKD approach not only facilitates the learning of a feature extractor with enhanced generalizability but also supports the development of personalized classifier heads that are tailored to the specific local data distributions of the clients. (2) **PFAKD v.s. FedPer & FedRep**: In FedPer and FedRep, the feature extractors trained on each client learn personalized feature information. In contrast, PFAKD locally learns both global and personalized feature information. (3) **PFAKD v.s. FedPAC & GPFL**: FedPAC/GPFL uses global feature centroid/global category embedding for each class to guide the training of feature extractors, but category-level features possess less global information. In contrast, PFAKD uses global feature information at the sample level to guide the training of the feature extractor, and features at the sample level have more global information.

More Local Epochs. The above experimental results show the superiority of our method for different rounds of iterations, and our method is more suitable for the case of the high number of local epochs. We show the test accuracy curves for the various methods for the three datasets at (local epochs)E=10, E = 20 in Fig. 3. To reduce communication overhead in FL, clients tend to increase the number of local iterations to reduce the number of communications and speed up convergence, but this tends to sacrifice the final accuracy of the model. From our experimental results, it can be seen that most of the algorithms show performance degradation as the local epoch increases (FedRep shows an increase in performance with the increase of local epoch on some datasets due to the low period of body training), whereas the performance of our method, PFAKD, shows very little or almost no degradation. The reason for this phenomenon is the more fine-grained feature guidance of PFAKD, which aligns the global and local features of each sample through knowledge distillation to introduce richer and more comprehensive global feature information to the local feature extractor, thereby improving the model's generalization and resulting in higher accuracy on the test set.

5 Conclusion

In this paper, we propose a new algorithm named PFAKD for PFL, designed to tackle the issue of data diversity within the FL framework. The core idea of PFAKD is achieving fine-grained feature alignment through a knowledge distillation-based approach. PFAKD enables the co-training of feature extractors and personalization of classifier heads by decoupling the models, allowing each client to optimize the model for its local data without sacrificing privacy and increasing the communication burden. Our experimental results show that PFAKD outperforms state-of-the-art methods in terms of the average testing accuracy of the model when dealing with heterogeneous data.

Acknowledgment. This work was partially supported by the National Natural Science Foundation of China (No. 62102131, 62306104, and 62180005), the Natural Science Foundation of Jiangsu Province of China (No. BK20210361 and BK20230949), the Provincial Key Research and Development Program of Hubei (No. 2023BAB065), China Postdoctoral Science Foundation (2023TQ0104), Jiangsu Excellent Postdoctoral Program (2023ZB140), and the Fundamental Research Funds for the Central Universities (No. B240201064).

References

1. Acar, D.A.E., Zhao, Y., Matas, R., Mattina, M., Whatmough, P., Saligrama, V.: Federated learning based on dynamic regularization. In: International Conference on Learning Representations (2020)
2. Arivazhagan, M.G., Aggarwal, V., Singh, A.K., Choudhary, S.: Federated learning with personalization layers. arXiv preprint arXiv:1912.00818 (2019)
3. Chen, D., Hu, J., Tan, V.J., Wei, X., Wu, E.: Elastic aggregation for federated optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12187–12197 (2023)
4. Chen, H.Y., Chao, W.L.: On bridging generic and personalized federated learning for image classification. In: International Conference on Learning Representations (2022)
5. Collins, L., Hassani, H., Mokhtari, A., Shakkottai, S.: Exploiting shared representations for personalized federated learning. In: International Conference on Machine Learning. pp. 2089–2099. PMLR (2021)
6. Fallah, A., Mokhtari, A., Ozdaglar, A.: Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. Advances in Neural Information Processing Systems **33**, 3557–3568 (2020)
7. Hsu, T.M.H., Qi, H., Brown, M.: Measuring the effects of non-identical data distribution for federated visual classification. arXiv preprint arXiv:1909.06335 (2019)
8. Jia, N., Qu, Z., Ye, B.: Communication-efficient federated learning via quantized clipped sgd. In: Wireless Algorithms, Systems, and Applications: 16th International Conference, WASA 2021, Nanjing, China, June 25–27, 2021, Proceedings, Part I 16. pp. 559–571. Springer (2021)
9. Kairouz, P., McMahan, H.B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A.N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al.: Advances and open problems in federated learning. Foundations and Trends® in Machine Learning **14**(1–2), 1–210 (2021)
10. Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S., Stich, S., Suresh, A.T.: Scaffold: Stochastic controlled averaging for federated learning. In: International Conference on Machine Learning. pp. 5132–5143. PMLR (2020)
11. Li, Q., He, B., Song, D.: Model-contrastive federated learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10713–10722 (2021)
12. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine Learning and Systems **2**, 429–450 (2020)
13. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial Intelligence and Statistics. pp. 1273–1282. PMLR (2017)
14. Qu, Z., Guo, S., Wang, H., Ye, B., Wang, Y., Zomaya, A.Y., Tang, B.: Partial synchronization to accelerate federated learning over relay-assisted edge networks. IEEE Transactions on Mobile Computing **21**(12), 4502–4516 (2021)
15. T Dinh, C., Tran, N., Nguyen, J.: Personalized federated learning with moreau envelopes. Advances in Neural Information Processing Systems **33**, 21394–21405 (2020)
16. Wang, J., Liu, Q., Liang, H., Joshi, G., Poor, H.V.: Tackling the objective inconsistency problem in heterogeneous federated optimization. Advances in Neural Information Processing Systems **33**, 7611–7623 (2020)

17. Wu, F., Guo, S., Qu, Z., He, S., Liu, Z., Gao, J.: Anchor sampling for federated learning with partial client participation. In: International Conference on Machine Learning. pp. 37379–37416. PMLR (2023)
18. Xu, J., Tong, X., Huang, S.L.: Personalized federated learning with feature alignment and classifier collaboration. In The Eleventh International Conference on Learning Representations (2023)
19. Ye, R., Xu, M., Wang, J., Xu, C., Chen, S., Wang, Y.: Feddisco: Federated learning with discrepancy-aware collaboration. In: International Conference on Machine Learning. pp. 39879–39902. PMLR (2023)
20. Yurochkin, M., Agarwal, M., Ghosh, S., Greenewald, K., Hoang, N., Khazaeni, Y.: Bayesian nonparametric federated learning of neural networks. In: International Conference on Machine Learning. pp. 7252–7261. PMLR (2019)
21. Zhang, J., Hua, Y., Wang, H., Song, T., Xue, Z., Ma, R., Cao, J., Guan, H.: Gpfl: Simultaneously learning global and personalized feature information for personalized federated learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5041–5051 (2023)
22. Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., Chandra, V.: Federated learning with non-iid data. arXiv preprint arXiv:1806.00582 (2018)