Bold but Cautious: Unlocking the Potential of Personalized Federated Learning through Cautiously Aggressive Collaboration

Xinghao Wu[1], Xuefeng Liu[1,2], Jianwei Niu *[1,2,3], Guogang Zhu[1], and Shaojie Tang[4]

[1]State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China
{wuxinghao, liu_xuefeng, niujianwei, buaa_zgg}@buaa.edu.cn
[2]Zhongguancun Laboratory, Beijing, China
[3]Zhengzhou University Research Institute of Industrial Technology, School of Information Engineering, Zhengzhou University, Zhengzhou, China
[4]Jindal School of Management, University of Texas at Dallas, USA
shaojie.tang@utdallas.edu

Abstract

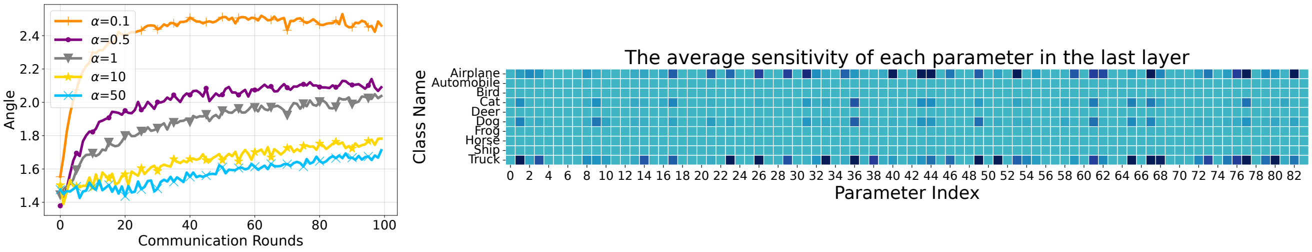
Personalized federated learning (PFL) reduces the impact of non-independent and identically distributed (non-IID) data among clients by allowing each client to train a personalized model when collaborating with others. A key question in PFL is to decide which parameters of a client should be localized or shared with others. In current mainstream approaches, all layers that are sensitive to non-IID data (such as classifier layers) are generally personalized. The reasoning behind this approach is understandable, as localizing parameters that are easily influenced by non-IID data can prevent the potential negative effect of collaboration. However, we believe that this approach is too conservative for collaboration. For example, for a certain client, even if its parameters are easily influenced by non-IID data, it can still benefit by sharing these parameters with clients having similar data distribution. This observation emphasizes the importance of considering not only the sensitivity to non-IID data but also the similarity of data distribution when determining which parameters should be localized in PFL. This paper introduces a novel guideline for client collaboration in PFL. Unlike existing approaches that prohibit all collaboration of sensitive parameters, our guideline allows clients to share more parameters with others, leading to improved model performance. Additionally, we propose a new PFL method named FedCAC, which employs a quantitative metric to evaluate each parameter's sensitivity to non-IID data and carefully selects collaborators based on this evaluation. Experimental results demonstrate that FedCAC enables clients to share more parameters with others, resulting in superior performance compared to state-of-the-art methods, particularly in scenarios where clients have diverse distributions. The code is integrated into our FL training framework: https://github.com/kxzxvbk/Fling.

1. Introduction

Federated Learning (FL) [25] enables institutions or devices to train a global model collaboratively without exposing raw data. It has been applied in many scenarios, such as computer vision [20, 5, 34], finance [21], healthcare [28, 29], etc. However, when the data among clients are non-independent and identically distributed (non-IID), the local tasks of clients are different, which brings a great challenge for training a global model for all clients [36, 7, 16]. Personalized federated learning (PFL) [6, 31, 17] allows each client to train a personalized model to focus on its data distribution. PFL aims to enable each client to get as much help as possible from other clients while minimizing the impact of non-IID data. A key question in PFL is to decide which parameters of a client should be localized to reduce the influence of non-IID or shared with others to get help.

To deal with this question, according to the characteristics of different neural network layers, the current mainstream work localizes layers sensitive to non-IID data while globally sharing the others. FedRep [4] and FedBN [18], for example, personalize classifier layers and batch normalization (BN) layers, respectively. The rationale behind these

*Corresponding author



(a)

(b)

Figure 1: Two preliminary experiments. (a) verifies that the influence of non-IID varies with time. (b) verifies that parameters in the same layer have different sensitivities.

methods is localizing parameters easily influenced by non-IID data to reduce the negative effect of collaboration.

However, although localizing sensitive layers can reduce the impact of non-IID data, these methods are too conservative for collaboration. The client can not fully utilize other clients' knowledge, and it is difficult to adapt to various complex non-IID scenarios. In addition, these methods are mostly empirical and heuristic and do not give a guideline for client collaboration.

We find that client collaboration depends not only on parameter sensitivity but also on variations in data distribution between clients. In non-IID scenarios, some clients still possess similar data distribution, allowing them to collaborate on sensitive parameters and benefit from each other. In this paper, we first propose a guideline for client parameter collaboration in PFL that takes both factors into account. It can be formulated as

$$U = I(\Psi \cdot \Omega). \tag{1}$$

In the above equation, $U \in \{0,1\}^N$ and $U_i = 1$ denotes that this parameter should not get contribution from client i. N denotes the total number of clients. $I(\cdot)$ is an indicator function. $\Psi \in [0,1]^N$, and Ψ_i indicates the data distribution difference between the current client and client i. A larger Ψ_i implies the two clients have less similar data distributions. $\Omega \in [0,1]$ indicates the current sensitivity of the parameter to non-IID data. The larger the Ω, the more sensitive it is to non-IID data. This guideline offers a cautiously aggressive collaboration. Even sensitive parameters (i.e., Ω is large) can still collaborate with clients with similar data distribution (i.e., Ψ_i is small), which enables a client can collaborate more with others while avoiding the negative effect of non-IID data. Current methods can also be incorporated under this guideline. For each client i, they set $\Psi_j = 1, i \neq j$ and select parameters sensitive to non-IID data in a layer-wise manner to set their $\Omega = 1$.

In practical use of the guideline, Ω is a critical factor that must be taken into account. Our research shows that two effects should be considered. First, Ω varies over time. In the early training stage, the impact of non-IID data is minimal, and parameters are less sensitive to non-IID data. As the training progresses, the impact of non-IID data becomes more pronounced, and parameters are more susceptible to its effects. To illustrate this, we conduct an experiment involving two clients in different non-IID scenarios and calculate the angle between their gradients in each round. As shown in Figure 1(a), in all non-IID scenarios (i.e., with all α values), the angle between the two gradients increases as the training progresses, indicating that the two clients are increasingly negatively influenced by each other. In other words, the impact of non-IID data grows over time.

Second, we discover that the current approach of selecting sensitive parameters layer-wise is too coarse-grained. The Ω value of parameters within the same layer can be different. To demonstrate this, we perform an experiment on the CIFAR-10 dataset with the CNN network. The proportion of samples of different classes in the training data is airplane:truck:cat:dog=4:4:1:1. We calculate the sensitivity of each parameter in the last layer of classifier, and the results are displayed in Figure 1(b). Each square represents the sensitivity of a parameter in the model's last layer, with darker colors indicating greater sensitivity. Each row in Figure 1(b) corresponds to parameters related to a specific class. The results reveal that the sensitivity of parameters in the same layer differs significantly. This indicates that we must select sensitive parameters at the parameter level to accomplish fine-grained collaboration. Additionally, this experiment demonstrates that the sensitivity of parameters is linked to data distribution. The parameters associated with the class that has more samples are more sensitive. This underscores the significance of considering differences in client data distribution (i.e., Ψ in Eq. (1)) when collaborating with sensitive parameters.

Building on the aforementioned guideline and observation, we introduce a new PFL method called FedCAC. FedCAC utilizes a sensitivity-based quantitative metric to access each parameter in each round, identifying the parameters sensitive to non-IID data as critical parameters and the rest as non-critical. All clients collaborate to train non-critical parameters, while FedCAC implements a time-varying collaboration strategy for critical parameters. Initially, each client receives assistance from a greater number of clients. As training progresses, it gradually collaborates only with clients that have more similar data distribution. Our primary contributions are summarized as follows:

- We propose a client collaboration guideline in PFL, which comprehensively considers the differences in data distribution between clients and the sensitivity of parameters to non-IID data. This framework enables clients to collaborate more aggressively while carefully avoiding the negative effect of non-IID data.
- Building upon the guideline, we propose a new PFL method that selects sensitive parameters in a parameter-wise manner and implements time-varying collaboration for them, allowing them to benefit fully from the assistance of other clients, thereby improving performance.
- The experimental results on CIFAR-10, CIFAR-100, and Tiny ImageNet prove that the performance of our method is significantly improved compared with the existing methods.

2. Related Work

PFL [33, 2, 3, 32] is a powerful means to address the non-IID problem in FL. The primary purpose of PFL is to enable personalized models to benefit from inter-client collaboration while mitigating the impact of non-IID. Among these advanced PFL methods, parameter decoupling and personalized aggregation are two representative methods.

Parameter Decoupling localizes some layers that are sensitive to non-IID data while sharing others globally. Some current studies choose critical layers based on the characteristics of the neural network, such as classifiers [1, 4], BN layers [18] or Adapter layers [27]. Some works also attempt to identify critical layers automatically via advanced techniques in deep learning, such as reinforcement learning [30] or hypernetwork [23]. These methods can effectively mitigate the negative impact of non-IID by letting localized layers adapt to the local task. Nonetheless, in these methods, the localized layers neglect to collaborate with others, resulting in poor performance in complex non-IID scenarios.

Personalized Aggregation lets clients with similar data distribution collaborate to alleviate the impact of non-IID data. Based on the attention mechanism, FedAMP [10] designs an attention-inducing function that allows similar models to aggregate with large weight. FedFomo [37] evaluates the degree of data distribution similarity through the model's validation set on the client side. Give better-performing models a larger aggregation weight. APPLE [22] learns a directed relational vector to guide personalized model aggregation. However, in practice, it is a challenge to find clients with similar data distribution to collaborate with, especially when the data distributions on clients vary dramatically.

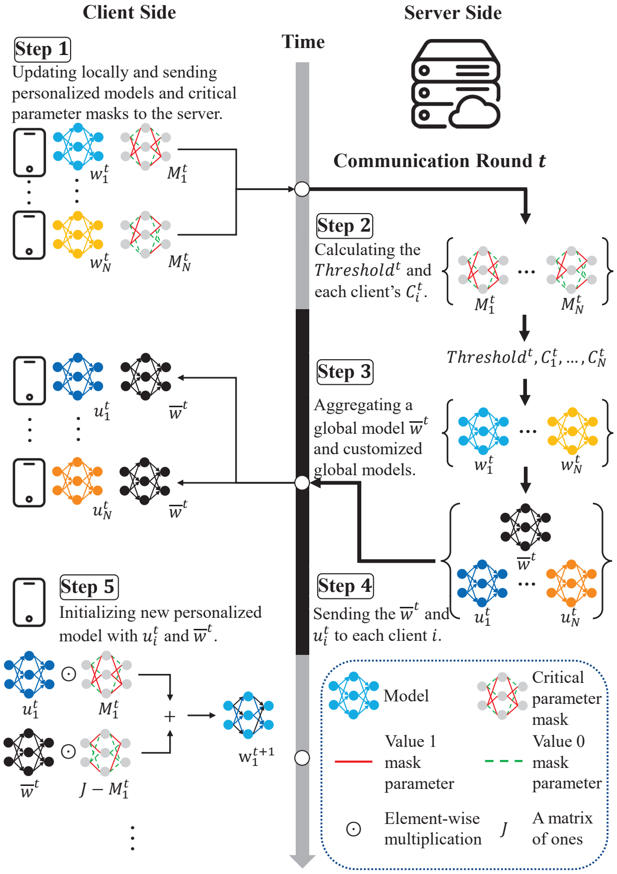

Client Side
Server Side
Time
Step 1
Updating locally and sending personalized models and critical parameter masks to the server.
w_1^t
M_1^t
w_N^t
M_N^t
Communication Round t
Step 2
Calculating the $Threshold^t$ and each client's C_i^t.
M_1^t
M_N^t
$Threshold^t, C_1^t, \ldots, C_N^t$
Step 3
Aggregating a global model $\overline{w}^t$ and customized global models.
w_1^t
w_N^t
u_1^t
$\overline{w}^t$
u_N^t
$\overline{w}^t$
$\overline{w}^t$
Step 4
Sending the $\overline{w}^t$ and u_i^t to each client i.
u_1^t
u_N^t
Step 5
Initializing new personalized model with u_i^t and $\overline{w}^t$.
u_1^t
M_1^t
$\odot$
+
w_1^{t+1}
$\overline{w}^t$
$J - M_1^t$
Model
Critical parameter mask
Value 1 mask parameter
Value 0 mask parameter
$\odot$ Element-wise multiplication
J A matrix of ones

Figure 2: The system workflow of FedCAC.

Briefly, current PFL methods only consider the influence of partial factors on collaboration and lack a comprehensive guideline for collaboration, which cannot adapt to various complex non-IID scenarios.

3. Method

3.1. Overview of FedCAC

Before starting, we first give an overview of FedCAC. As shown in Figure 2, the training process in each communication round of FedCAC can be summarized as follows:

Step 1: Each client i trains the personalized model w_i^t and selects critical parameters to generate a critical parameter mask M_i^t. Then, it sends the w_i^t and M_i^t to the server.

Step 2: The server calculates a threshold for critical parameter collaboration based on the overlap ratio of the locations of the critical parameters of different clients. Then, it selects a set of clients C_i^t to collaborate critical parameters for each client i based on the threshold.

Step 3: The server aggregates personalized models to generate a global model $\overline{w}^t$ and a customized global model u_i^t for each client i based on C_i^t.

Step 4: The server sends $\overline{w}^t$ and u_i^t to each client i.

Step 5: Each client i initializes the personalized model w_i^{t+1} based on $\overline{w}^t$, u_i^t and M_i^t.

In the following, we illustrate the details of FedCAC. Step 1 is introduced in section 3.3. Steps 2 to 5 are described in section 3.4.

3.2. Problem Definition of PFL

Traditional FL trains a global model for all clients, whereas PFL seeks to train a personalized model w_i for each client i that performs optimally on their respective local data distribution D_i. Let $L_i(w_i; D_i)$ denote the loss function of client i, then the training objective of PFL can be formulated as

$$\min_{w_1, w_2, \ldots, w_N} \sum_{i=1}^{N} L_i(w_i; D_i), \tag{2}$$

where N is the total number of clients in FL. One approach to optimizing the objective described above is to utilize the local data on each client to minimize their corresponding loss function $L_i(w_i; D_i^{train})$, where the D_i^{train} is the local data distribution of client i. However, clients participating in FL typically have very limited local data. As a result, the model can easily overfit to the D_i^{train}, resulting in poor generalization of w_i on D_i. Collaborating with personalized models of other clients in the training process is essential. However, in non-IID scenarios, different clients have distinct data distributions (i.e., $D_i \neq D_j, i \neq j$), making collaboration with other clients potentially harmful to the performance of w_i on their local tasks. Therefore, how to reduce the impact of non-IID when clients collaborate is a challenge in PFL.

3.3. Sensitivity Based Parameter Selection

As discussed in Eq. (1), identifying parameters that are sensitive to non-IID (i.e., Ω in Eq. (1)) is a critical aspect of collaboration design. Ω varies among different parameters and at different stages of training. We decouple these two factors in our approach. In this section, we begin by assessing the sensitivity of various parameters and selecting the most sensitive subset of parameters to non-IID data. In section 3.4, we examine the impact of the training stage when choosing collaborative clients for sensitive parameters.

In FedCAC, we employ a sensitivity-based approach for parameter selection. As discussed in previous research [26, 15, 38], the sensitivity of a parameter refers to the degree of variation in the model output or loss function when the parameter is set to 0. Specifically, given a model w whose parameter set is expressed as $\Theta = \{\theta_1, ..., \theta_i, ..., \theta_n\}$, then the sensitivity of the i-th parameter is defined as

$$s_i = |L(\Theta) - L(\theta_1, ..., \theta_{i-1}, 0, \theta_{i+1}, ..., \theta_n)|, \tag{3}$$

where L is the loss function. The sensitivity of a parameter is a useful metric for quantifying its significance to the local task. Furthermore, parameters crucial to the local task are more likely to degrade the model's accuracy when influenced by clients with different data distributions. As a result, the sensitivity of a parameter on the local task can also serve as a measure of the parameter's sensitivity to non-IID data.

However, it can be seen from Eq. (3) that evaluating each parameter requires an additional forward propagation, which significantly increases the computation cost. To solve this problem, we use the first-order Taylor approximation to substitute the calculation of s_i:

$$\begin{aligned} s_i &= |L(\Theta) - L(\theta_1, ..., \theta_{i-1}, 0, \theta_{i+1}, ..., \theta_n)| \\ &= |\nabla_{\theta_i} L(\Theta) \cdot \theta_i + R_1(\Theta)| \\ &\approx |\nabla_{\theta_i} L(\Theta) \cdot \theta_i|. \end{aligned} \tag{4}$$

The gradients of all parameters can be obtained by one back propagation, which greatly reduces the computation overhead of s_i.

In FL, the client generally updates the model locally for several epochs. A simple way to apply Eq. (4) to PFL is by computing sensitivity only in the final epoch. However, this approach may be prone to significant randomness. To overcome this limitation, we replace $\nabla_{\theta_i} L(\Theta)$ with the variation of model $\Delta\theta_i^t = \theta_i^{t,E} - \theta_i^{t,0}$, so as to calculate the sensitivity of each parameter to non-IID accurately. Here, the $\theta_i^{t,j}$ represents θ_i after the j-th ($j \in [1, E]$) local epoch in round t ($t \in [1, T]$). E is the maximum local epoch. T is the maximum communication round. Based on this, our parameter sensitivity measurement metric can be expressed as

$$s_i^t = |\Delta\theta_i^t \cdot \theta_i^{t,E}|. \tag{5}$$

In each training round t of FedCAC, each client $i \in [1, N]$ minimizes local loss function (e.g., cross-entropy loss in classification task) on local data by E epochs to update personalized model w_i^t. Next, it computes the sensitivity of each parameter by Eq. (5) and constructs the parameter sensitivity matrix S_i^t. Based on S_i^t, each client selects the most sensitive parameters as *critical* parameters, and the remaining parameters are considered *non-critical*. We utilize a hyperparameter $\tau \in [0, 1]$ to control the proportion of critical parameters to the total parameters. Because the distribution of parameter values in different neural network layers is different, critical parameters are selected layer by layer. That is, parameters with the top-τ sensitive in each layer are identified as critical parameters. We use a mask matrix M_i^t to mark the critical parameters of client i. Each element

$$m_{j,k}^t = \begin{cases} 1, & \text{if } S_{i,j,k}^t \text{ in the top-}\tau \text{ largest of } S_{i,j}^t \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

in M_i^t marks whether the k-th parameter of the j-th layer is a critical parameter. $S_{i,j}^t$ denotes the sensitivity set of parameters at the j-th layer of the model of client i. $S_{i,j,k}^t$ denotes the k-th item of $S_{i,j}^t$.

It should be noted that by Eq. (6), the M_i^t is a binary mask. It only needs 1 bit to store each element. The transmission cost of a mask is about 3% compared to transmitting a model (32-bit per element). Therefore, the mask matrix introduced by FedCAC does not significantly increase the client's storage overhead and uplink communication cost.

3.4. Parameter Collaboration in FedCAC

Once we decouple the critical and non-critical parameters, we proceed to design collaboration strategies for them on the server. According to Eq. (1), non-critical parameters' Ω are quite small, and they can safely collaborate with all clients. Conversely, critical parameters have a large Ω and need to collaborate with clients that have similar data distributions carefully. Furthermore, as illustrated in Figure 1(a), critical parameters can collaborate with more clients in the early training stage and gradually collaborate with clients whose data distributions are more similar. However, due to the privacy constraints in FL, we can not know the data distribution of clients. As shown in the supplementary material, we find that the overlap ratio of client critical parameter locations can indicate the similarity degree of client data distribution, which can help us know the Ψ in Eq. (1).

With the above idea, we design a time-varying collaboration strategy for critical parameters. We utilize $O_{i,j}^t = \frac{||M_i^t - M_j^t||_1}{2n}$ to represent the overlap ratio of critical parameter locations of the client i and client j in round t, where $||\cdot||_1$ denotes the L_1 norm and n is the total number of parameters. In each training round t, after receiving M_i^t from each client, the server calculates a threshold:

$$\text{Threshold}^t = O_{avg}^t + \frac{t}{\beta} \times (O_{max}^t - O_{avg}^t), \quad (7)$$

where $O_{avg}^t = \frac{1}{N\cdot(N-1)} \sum_{i\neq j} O_{i,j}^t$ and $O_{max}^t = \max_{i\neq j}\{O_{i,j}^t\}$. The extent of assistance provided to the critical parameters by other clients is determined by $\beta \in [1, T]$. The higher the β, the more substantial the help provided to the critical parameters. Based on the Threshold^t, the server selects a set of collaboration clients for each client to collaborate critical parameters by

$$C_i^t = \{j | O_{i,j}^t \geq \text{Threshold}^t, j \neq i\}. \quad (8)$$

As the training progresses, the value of Threshold^t gradually rises, resulting in a reduction in $|C_i^t|$. This implies that the client i collaborates only with clients whose data distribution is more similar. Once $t > \beta$, each client trains critical parameters independently.

After obtaining the set of collaborative clients C_i^t, the server first aggregates a global model

$$\overline{w}^t = \frac{1}{N} \sum_{i=1}^{N} w_i^t \quad (9)$$

for all clients to collaborate on non-critical parameters. It also aggregates a customized global model

$$u_i^t = \frac{1}{|C_i^t| + 1} \sum_{j \in C_i^t \cup \{i\}} w_j^t, \quad (10)$$

for each client i to collaborate on critical parameters. After receiving $\overline{w}^t$ and u_i^t from the server, the client i initializes the personalized model w_i based on

$$w_i^{t+1} = u_i^t \odot M_i^t + \overline{w}^t \odot (J - M_i^t), \quad (11)$$

where J is a matrix with every element equal to one. In this way, FedCAC allows clients to aggressively get help from more clients, while carefully reducing the impact of non-IID data by taking into account parameter sensitivity, training stages, and differences in data distribution.

The details of the training process are summarized as pseudocode in Algorithm 1.

Algorithm 1 FedCAC

```
Input: Each client's initial personalized model $w_i^1$; Number of clients $N$; Total communication round $T$; Local epoch number $E$; Hyperparameters $\tau$, $\beta$.
Output: Personalized model $w_i^T$ for each client.
for $t = 1$ to $T$ do
    Client-side:
    for $i = 1$ to $N$ in parallel do
        Updating $w_i^t$ for $E$ local epochs.
        Evaluating the sensitivity of each parameter by Eq .(5).
        Obtaining critical parameter mask matrix $M_i^t$ by Eq. (6).
        Sending $w_i^t$ and $M_i^t$ to the server.
    end for
    Server-side:
    Calculating $C_i^t$ for each client $i$ by Eq. (7) and Eq. (8).
    Aggregating a global model $\overline{w}^t$ by Eq. (9).
    Aggregating a $u_i^t$ for each client $i$ by Eq. (10).
    Sending $\overline{w}^t$ and $u_i^t$ to each client $i$.
    Client-side:
    for $i = 1$ to $N$ in parallel do
        Initializing $w_i^{t+1}$ with $\overline{w}^t$ and $u_i^t$ by Eq. (11).
    end for
end for
```

4. Experiments

4.1. Dataset Settings

We conduct experiments on three datasets, including CIFAR-10 [13], CIFAR-100 [12], and Tiny ImageNet [14]. Meanwhile, to verify the effectiveness of our method in different scenarios, we adopt two commonly used non-IID scenarios for experiments. One is pathological non-IID, and the other is Dirichlet non-IID.

Pathological non-IID is one of the most common non-IID settings in FL proposed by [24]. In this setting, each client is randomly assigned two classes of data, and the data distributions among clients are very different.

Dirichlet non-IID is also a commonly used non-IID setting in FL [9, 19, 11, 35]. In this setting, each client's data are drawn from the Dirichlet distribution, with $q \sim Dir(\alpha p)$, where p is the prior distribution of all classes, and α is a hyperparameter used to control the degree of non-IID. As the α increases, the class imbalance degree of each client gradually decreases, resulting in more challenging local tasks (i.e., an increase in the number of classes and a decrease in the sample size for each class). At the same time, the difference in data distribution among clients gradually decreases while the variety of client data distribution increases. Therefore, Dirichlet non-IID is an effective way to evaluate the performance of methods in various and complex non-IID scenarios. To aid in intuitive understanding, we provide a visualization of the data partitioning in the supplementary material.

To highlight the effectiveness of the client collaboration, we assign a small amount of data to each client. Each client has 500 training samples and 100 test samples. Test data and training data have the same data distribution.

4.2. Implementation Details

We evaluate FedCAC against four SOTA methods, namely FedPer [1], FedRep [4], FedBN [18], and FedAMP [10]. For the hyperparameters specific to all methods, we follow the optimal hyperparameter combination stated in their paper. In our methods, we adopt the SGD optimizer with a learning rate of 0.1. For federated learning training-related hyperparameters, we set the number of clients $N = 40$, the number of local epochs $E = 5$, and batch size equals 100. The experiments are conducted with the following settings: the maximum communication round T is set to 500 to ensure full convergence. In each run, we evaluate the uniform averaging test accuracy across all clients in each communication round and select the best accuracy as the final result. The ResNet [8] network structure is used, specifically ResNet-8 for CIFAR-10, and ResNet-10 for CIFAR-100 and Tiny ImageNet. It should be noted that the BN layer is included in ResNet. The Statistical parameters *running_mean* and *running_var* in the BN layers cannot be used to calculate sensitivity by Eq. (5). Following the work in [18], we set these parameters as critical parameters. Each experiment is repeated five times, and the mean and standard deviation are reported.

Methods	CIFAR-10	CIFAR-100	Tiny
FedAvg	54.33 ± 3.03	25.68 ± 1.31	17.93 ± 2.81
Separate	85.85 ± 0.93	88.97 ± 1.10	86.43 ± 1.08
FedAMP	88.88 ± 0.83	91.80 ± 0.61	88.23 ± 0.52
FedRep	87.10 ± 0.91	90.25 ± 0.60	87.22 ± 0.94
FedBN	87.02 ± 1.41	91.74 ± 0.62	89.41 ± 0.32
FedPer	87.51 ± 0.95	90.33 ± 0.60	87.17 ± 1.10
FedCAC	**89.77 ± 1.14**	**93.05 ± 0.90**	**90.36 ± 0.75**

Table 1: Comparison results under Pathological non-IID.

4.3. Comparison with SOTA Methods

Pathological non-IID. In the pathological non-IID scenario, each client is given only two classes of data and performs a simple binary classification task locally. From 'Separate' in Table 1, we can see that the client can obtain high accuracy by training itself. However, data distribution among clients varies greatly, making collaboration a challenge. Direct collaboration with other clients can actually have a negative impact, resulting in poor performance for FedAvg. FedRep, FedBN, and FedPer perform better by personalizing the layers that are sensitive to non-IID data, thereby mitigating the negative impact. FedAMP selects clients with similar data distributions to collaborate, which reduces the impact of non-IID data and leads to better model performance. However, as the dataset difficulty increases (from CIFAR-10 with 10 classes to Tiny Imagenet with 200 classes), it becomes harder for clients to find similar clients for collaboration, which reduces FedAMP's advantage. Our approach, which allows sensitive parameters to receive help from clients with similar data distributions and insensitive parameters to receive help globally, overcomes this limitation and achieves the best performance across all datasets.

Dirichlet non-IID. We also evaluate our approach in the Dirichlet non-IID scenario. We conduct experiments with α values of $\{0.1, 0.5, 1.0\}$ for CIFAR-10 and $\{0.01, 0.1, 0.5\}$ for CIFAR-100 and Tiny Imagenet.

Based on the experimental results shown in Table 2, it can be observed that all SOTA methods achieve significant performance improvements when compared to the 'Separate' approach in the Dirichlet non-IID scenario. Among these methods, FedAMP and FedPer show more obvious superiority on CIFAR-10. This can be attributed to the fact that it is easier for clients to find and collaborate with other clients having similar data distributions in the 10-classification dataset, thereby resulting in better model per-

	CIFAR-10			CIFAR-100			Tiny Imagenet		
Method	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 1.0$	$\alpha = 0.01$	$\alpha = 0.1$	$\alpha = 0.5$	$\alpha = 0.01$	$\alpha = 0.1$	$\alpha = 0.5$
FedAvg	54.36±2.73	60.41±1.36	60.91±0.72	27.83±3.03	34.16±0.70	32.78±0.23	20.56±0.25	21.26±1.28	20.32±0.91
Separate	81.45±1.60	60.15±0.86	52.24±0.41	82.91±0.96	45.57±0.90	22.65±0.51	64.47±3.11	24.07±0.62	8.75±0.30
FedAMP	84.99±1.82	68.26±0.79	64.87±0.95	85.11±1.15	46.68±1.06	24.74±0.58	69.86±2.03	27.85±0.71	10.70±0.32
FedRep	84.59±1.58	67.69±0.86	60.52±0.72	84.40±1.43	51.25±1.37	26.97±0.33	70.45±1.11	30.83±1.05	12.14±0.28
FedBN	83.55±2.32	66.79±1.08	62.20±0.67	85.43±0.87	54.35±0.63	36.94±0.94	73.90±1.45	33.34±0.71	19.61±0.35
FedPer	84.43±0.47	68.80±0.49	64.92±0.66	83.35±0.78	51.38±0.94	28.25±1.03	72.63±0.78	32.33±0.31	12.69±0.42
FedCAC	**86.82±1.18**	**69.83±0.46**	**65.39±0.51**	**86.58±0.85**	**57.22±1.52**	**38.64±0.63**	**75.76±1.21**	**40.19±1.20**	**23.70±0.28**

Table 2: Comparison results under Dirichlet non-IID on CIFAR-10, CIFAR-100, and Tiny Imagenet.

formance for FedAMP. Moreover, the 10-classification task also makes the classifier training simple, allowing the client to train the classifier effectively while reducing the influence of other clients, resulting in the better model performance of FedPer.

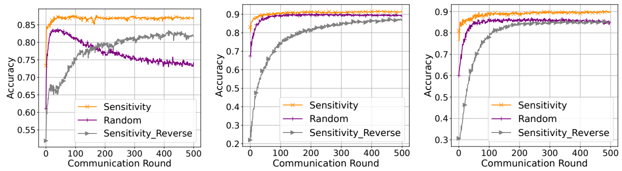
The experimental results on CIFAR-100 and Tiny Imagenet datasets show that the performance of SOTA methods differs significantly from that on CIFAR-10. With the increase in the number of classes, the variety of client data distribution increases, making it harder for FedAMP to find clients with similar data distributions, resulting in its models hardly getting help from others. The increase in the number of classes can also make local tasks harder, causing difficulty for clients to train their classifiers and increasing the risk of overfitting. Therefore, the superiority of FedPer reduces. Interestingly, most SOTA methods perform poorly when $\alpha = 0.5$. Moreover, no SOTA method is better than FedAvg on the Tiny Imagenet dataset. This suggests that the current PFL approach does not fully leverage the knowledge of other clients when collaborating, leading to poor performance in the face of difficult datasets and complex client-side data distribution scenarios.

FedCAC can obtain the best model performance under all settings. It performs particularly well when the clients' data distributions are complex and their local tasks are difficult (e.g., $\alpha = 0.1$ or $\alpha = 0.5$ in Tiny Imagenet). This demonstrates FedCAC can adapt to a variety of complex non-IID scenarios and achieve more efficient collaboration by fully considering the client data distribution differences, training stages, and parameter sensitivity. Moreover, it confirms the effectiveness of the guideline proposed in Eq. (1).

4.4. Component Analysis

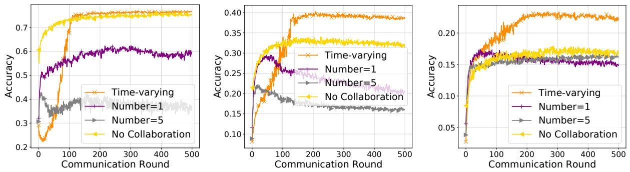
In this section, we verify the effect of various components introduced to FedCAC.

Effect of Critical Parameter Selection Schemes. We compare three critical parameter selection methods: 'Sensitivity', which is based on sensitivity, 'Random', which randomly selects some parameters as critical parameters in each round, and 'Sensitivity_Reverse', which uses our sensitivity-based method to choose insensitive parameters as the critical parameters. We conduct experiments on Pathological non-IID with hyperparameter $\tau = 0.5$. During training, we localize all critical parameters to exclude the influence of critical parameter collaboration.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny Imagenet

Figure 3: Effect of different critical parameter selection schemes on CIFAR-10, CIFAR-100, and Tiny Imagenet datasets.



(a) $\alpha = 0.01$ (b) $\alpha = 0.1$ (c) $\alpha = 0.5$

Figure 4: Effect of different critical parameter collaboration schemes on Tiny Imagenet.

The results are shown in Figure 3. We can see that the accuracy of 'Random' increases fast in the early stage, which indicates that it randomly selects some sensitive parameters and reduces the influence of non-IID data. However, its accuracy decreases as the training goes on, indicating that the influence of non-IID data is gradually increasing. The 'Sensitivity_Reverse' method results in slow convergence and poor final accuracy due to the client having a greater influence on the local task when collaborating. The 'Sensitivity' method is superior to other methods in all datasets, in terms of convergence speed and accuracy, which indicates that our method can efficiently and accurately select parameters sensitive to non-IID data.

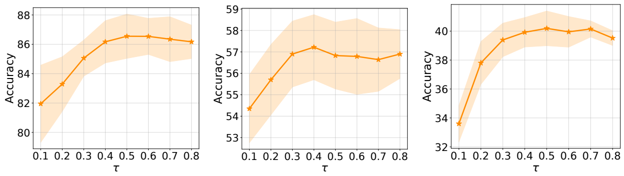
METHODS	CIFAR-10	CIFAR-100	TINY
ALL	86.82±1.18	57.22±1.52	40.19±1.20
AS CRITICAL	85.42±1.39	55.15±1.10	37.97±0.63

Table 3: The effect of non-critical parameters collaboration on three datasets.

Effect of Critical Parameter Collaboration Schemes. To demonstrate the effectiveness and necessity of our critical parameter collaboration approach in various scenarios, we conduct experiments on the challenging Tiny Imagenet dataset under the Dirichlet non-IID setting, where we sample $\alpha \in \{0.01, 0.1, 0.5\}$. We introduce two additional methods for comparison, aiming at highlighting the challenges involved in designing collaborative approaches for critical parameters. 'Number=1' and 'Number=5' represent methods where each client collaborates with the one and five clients with the highest overlap rate of critical parameter locations during the training process. As a baseline, we include the 'No Collaboration' method, which localizes the client's critical parameters.

The experimental results are shown in Figure 4. When $\alpha = 0.01$, there are only a few classes of data on each client, and the local task is relatively simple. The client itself training critical parameters can achieve better results. We can see that 'No Collaboration' works well. In this case, the non-IID degree is high, and client-side collaboration can have a negative impact. We can see that 'Number=1' and 'Number=5' have a significant performance degradation compared to 'No Collaboration'. And the more clients collaborate, the more obvious the decline. Although the 'Time-varying' designed by us encourages clients to collaborate in the early training stage, resulting in a lower convergence rate than 'No Collaboration', the critical parameters of the clients can also provide positive help to each other in the end, thus achieving better performance. Meanwhile, we can see that with the increase of α, the local task of the client gradually becomes more difficult, and the client itself cannot train the critical parameters well. At this time, the client is more inclined to seek help from other clients, and the benefits of collaboration are also increasing. Compared with 'No Collaboration', our 'Time-varying' has increased performance more and more.

The above experiments illustrate that the collaboration of critical parameters is a huge challenge. It is not only necessary to consider the client data distribution similarity but also to consider the influence of non-IID in different training stages. This also shows that based on our guidelines, our approach can select the right collaboration clients at the right time, further improving the model's performance.



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny Imagenet

Figure 5: Effect of τ in Dirichlet non-IID scenario with $\alpha = 0.1$.

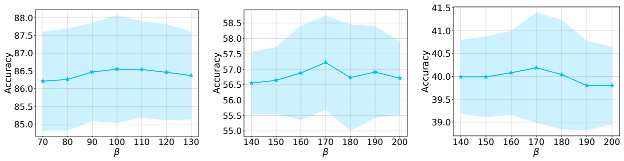
Effect of Collaboration on Non-critical Parameters. To validate the idea that non-critical parameters, which are less sensitive to non-IID, can benefit from collaborating with all clients, we conduct experiments in the Dirichlet non-IID scenario with $\alpha = 0.1$.

The experimental results in Table 3 demonstrate that collaborating on non-critical parameters with all clients (denoted by 'All') yields better performance than treating non-critical parameters like critical parameters (denoted by 'As Critical') on all three datasets. This suggests that collaborating on non-critical parameters with all clients can provide more benefits than only collaborating with similar data distributions. These results support our guideline that non-critical parameters, which are less sensitive to non-IID, can collaborate with as many clients as possible for more help.

4.5. Effect of Hyperparameters in FedCAC

Effect of τ. τ is a hyperparameter used to control the proportion of critical parameters. If τ is too small, some sensitive parameters may not be selected, and the collaboration among clients will be affected by non-IID, whereas a large value of τ may lead to a lack of help for insensitive parameters. We conduct the experiment in Dirichlet non-IID scenario with $\alpha = 0.1$ to evaluate this intuition, and the experimental results are shown in Figure 5. We observe that when the τ is small, the accuracy is very low, indicating that client collaboration significantly impacts each other's local task accuracy. As τ increases, the influence among clients gradually decreases, and the accuracy gradually increases. The accuracy is highest when τ equals around 0.5. When τ is increased again, the help clients get from other clients decreases, and the accuracy gradually decreases. This is in line with our expectations. Furthermore, we find that near the optimal τ, the accuracy does not change much when the τ is changed slightly, indicating that FedCAC is robust to τ. For practical use, we preferentially select $\tau = 0.5$ and slightly adjust on this basis in different non-IID scenarios.

Effect of β. The hyperparameter β controls the degree to which critical parameters of clients receive help from other clients. To evaluate the impact of β, we conduct the experiment in Dirichlet non-IID scenario with $\alpha = 0.1$, and the experimental results are shown in Figure 6. A low value of β means that clients receive little help from others, leading



(a) CIFAR-10 (b) CIFAR-100 (c) Tiny Imagenet

Figure 6: Effect of β in Dirichlet non-IID scenario with $\alpha = 0.1$.

to low accuracy. On the other hand, when β is high, clients cannot timely reduce the number of collaborating clients, resulting in a greater influence of non-IID. It is observed that the accuracy is less sensitive to slight variations in β near its optimal value, indicating that FedCAC is robust to β. In this paper, for the experiments on CIFAR-10, CIFAR-100, and Tiny Imagenet, we preferentially select $\beta = 100$, $\beta = 170$, and $\beta = 170$, respectively, and make minor adjustments on this basis in different non-IID scenarios.

5. Conclusion

In this paper, we propose a comprehensive guideline for PFL collaboration that takes into account both the data distribution difference between clients and the sensitivity of each parameter. Building on this guideline, we introduce a novel PFL method called FedCAC, which leverages a quantitative metric to evaluate the sensitivity of each parameter and dynamically selects clients with similar data distribution for the sensitive parameters to collaborate. Our experimental results demonstrate that guided by the proposed guideline, our FedCAC method effectively makes each client obtain more help from others, resulting in superior performance in various complex non-IID scenarios.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No. 61976012.

References

[1] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019. 3, 6

[2] Fengwen Chen, Guodong Long, Zonghan Wu, Tianyi Zhou, and Jing Jiang. Personalized federated learning with a graph. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2575–2582. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track. 3

[3] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. In *International Conference on Learning Representations*, 2022. 3

[4] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. *arXiv preprint arXiv:2102.07078*, 2021. 1, 3, 6

[5] Keval Doshi and Yasin Yilmaz. Federated learning-based driver activity recognition for edge devices. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3338–3346, 2022. 1

[6] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020. 1

[7] Xuan Gong, Abhishek Sharma, Srikrishna Karanam, Ziyan Wu, Terrence Chen, David Doermann, and Arun Innanje. Ensemble attention distillation for privacy-preserving federated learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15076–15086, 2021. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[9] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019. 6

[10] Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021. 3, 6

[11] Jinkyu Kim, Geeho Kim, and Bohyung Han. Multi-level branched regularization for federated learning. In *International Conference on Machine Learning*, pages 11058–11073. PMLR, 2022. 6

[12] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[13] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). *URL http://www. cs. toronto. edu/kriz/cifar. html*, 5, 2010. 6

[14] Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015. 6

[15] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. SNIP: SINGLE-SHOT NETWORK PRUNING BASED ON CONNECTION SENSITIVITY. In *International Conference on Learning Representations*, 2019. 4

[16] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021. 1

[17] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pages 6357–6368. PMLR, 2021. 1

[18] Xiaoxiao Li, Meirui JIANG, Xiaofei Zhang, Michael Kamp, and Qi Dou. FedBN: Federated learning on non-IID features via local batch normalization. In *International Conference on Learning Representations*, 2021. 1, 3, 6

[19] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *NeurIPS*, 2020. 6

[20] Yang Liu, Anbu Huang, Yun Luo, He Huang, Youzhi Liu, Yuanyuan Chen, Lican Feng, Tianjian Chen, Han Yu, and Qiang Yang. Fedvision: An online visual object detection platform powered by federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13172–13179, 2020. 1

[21] Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. Federated learning for open banking. In *Federated Learning: Privacy and Incentive*, pages 240–254. Springer, 2020. 1

[22] Jun Luo and Shandong Wu. Adapt to adaptation: Learning personalization for cross-silo federated learning. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2166–2173. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track. 3

[23] Xiaosong Ma, Jie Zhang, Song Guo, and Wenchao Xu. Layer-wised model aggregation for personalized federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10092–10101, 2022. 3

[24] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017. 6

[25] H Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. *arXiv preprint arXiv:1602.05629*, 2016. 1

[26] Pavlo Molchanov, Arun Mallya, Stephen Tyree, Iuri Frosio, and Jan Kautz. Importance estimation for neural network pruning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11256–11264, 2019. 4

[27] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022. 3

[28] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):119, 2020. 1

[29] Adam Sadilek, Luyang Liu, Dung Nguyen, Methun Kamruzzaman, Stylianos Serghiou, Benjamin Rader, Alex Ingerman, Stefan Mellem, Peter Kairouz, Elaine O Nsoesie, et al. Privacy-first health research with federated learning. *NPJ digital medicine*, 4(1):132, 2021. 1

[30] Benyuan Sun, Hongxing Huo, Yi Yang, and Bo Bai. Partialfed: Cross-domain personalized federated learning via partial initialization. *Advances in Neural Information Processing Systems*, 34:23309–23320, 2021. 3

[31] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020. 1

[32] Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. Fedproto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8432–8440, 2022. 3

[33] Xueyang Tang, Song Guo, and Jingcai Guo. Personalized federated learning with contextualized generalization. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2241–2247. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track. 3

[34] Chunnan Wang, Xiang Chen, Junzhe Wang, and Hongzhi Wang. Atpfl: Automatic trajectory prediction model design under federated learning framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6563–6572, 2022. 1

[35] Xinghao Wu, Jianwei Niu, Xuefeng Liu, Tao Ren, Zhangmin Huang, and Zhetao Li. pfedgf: Enabling personalized federated learning via gradient fusion. In *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 639–649. IEEE, 2022. 6

[36] Lin Zhang, Yong Luo, Yan Bai, Bo Du, and Ling-Yu Duan. Federated learning for non-iid data via unified feature learning and optimization objective alignment. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4420–4428, 2021. 1

[37] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization. In *International Conference on Learning Representations*, 2021. 3

[38] Yupeng Zhang, Hongzhi Zhang, Sirui Wang, Wei Wu, and Zhoujun Li. PATS: Sensitivity-aware noisy learning for pretrained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3680–3687, Abu Dhabi, United Arab Emirates, Dec. 2022. Association for Computational Linguistics. 4