Rethinking Personalized Client Collaboration in Federated Learning

Leijie Wu, *Student Member, IEEE*, Song Guo, *Fellow, IEEE*, Yaohong Ding, *Student Member, IEEE*, Junxiao Wang, Wenchao Xu, *Member, IEEE*, Yufeng Zhan, and Anne-Marie Kermarrec

***Abstract*—Federated Learning (FL) has gained considerable attention recently, as it allows clients to cooperatively train a global machine learning model without sharing raw data. However, its performance can be compromised due to the high heterogeneity in clients' local data distributions, commonly known as Non-IID (non-independent and identically distributed). Moreover, collaboration among highly dissimilar clients exacerbates this performance degradation. Personalized FL seeks to mitigate this by enabling clients to collaborate primarily with others who have similar data characteristics, thereby producing personalized models. We noticed that existing methods for assessing model similarity often do not capture the genuine relevance of client domains. In response, our paper enhances personalized client collaboration in FL by introducing a metric for domain relevance between clients. Specifically, to facilitate optimal coalition formation, we measure the marginal contributions of client models using coalition game theory, providing a more accurate representation of potential client domain relevance within the FL privacy-preserving framework. Based on this metric, we then adjust each client's coalition membership and implement a personalized FL aggregation algorithm that is robust to Non-IID data domain. We provide a theoretical analysis of the algorithm's convergence and generalization capabilities. Our extensive evaluations on multiple datasets, including MNIST, Fashion-MNIST, CIFAR-10, and CIFAR-100, and under varying Non-IID data distributions (Pathological and Dirichlet), demonstrate that our personalized collaboration approach consistently outperforms contemporary benchmarks in terms of accuracy for individual clients.**

***Index Terms*—Coalition game theory, multiwise collaboration, personalized federated learning, shapley value.**

Manuscript received 16 September 2023; revised 13 March 2024; accepted 22 April 2024. Date of publication 2 May 2024; date of current version 5 November 2024. This work was supported by the Key-Area Research and Development Program of Guangdong Province under Grant 2021B0101400003, in part by Hong Kong RGC Research Impact Fund under Grant R5060-19, and Grant R5034-18, in part by Areas of Excellence Scheme under Grant AoE/E-601/22-R, in part by General Research Fund under Grant 152203/20E, Grant 152244/21E, Grant 152169/22E, and Grant 152228/23E, in part by Shenzhen Science and Technology Innovation Commission under Grant JCYJ20200109142008673, and in part by National Natural Science Foundation of China under Grant 62102022. Recommended for acceptance by L. Duan. *(Corresponding authors: Song Guo; Yufeng Zhan.)*

Leijie Wu, Yaohong Ding, and Wenchao Xu are with the The Hong Kong Polytechnic University, Hong Kong (e-mail: lei-jie.wu@connect.polyu.hk; yaohong.ding@connect.polyu.hk; wenchao.xu@polyu.edu.hk).

Song Guo is with the The Hong Kong University of Science and Technology, Hong Kong (e-mail: songguo@cse.ust.hk).

Junxiao Wang is with the Guangzhou University, Guangdong 511370, China (e-mail: wangjunxiao@live.com).

Yufeng Zhan is with the Beijing Institute of Technology, Beijing 100811, China (e-mail: yu-feng.zhan@bit.edu.cn).

Anne-Marie Kermarrec is with the École Polytechnique Fédérale de Lausanne (EPFL), 1015 Lausanne, Switzerland (e-mail: anne-marie.kermarrec@epfl.ch).

Digital Object Identifier 10.1109/TMC.2024.3396218

I. Introduction

With the ongoing advancement of web services, vast amounts of client data are generated daily, that can be immediately exploited through machine learning technology. Indeed, machine learning models, when fueled by such extensive data, have found applications in a myriad of contexts, revolutionizing fields like precision medicine and recommendation systems, to name a few. Within these applications, the precision and generalizability of models are paramount, attributes that are enhanced by training on large data volumes. However, legal constraints, business confidentiality, and individual privacy concerns prevent clients from directly sharing data. This leads to the creation of "data silos", limiting the potential enhancement of model capabilities [1], [2].

Federated Learning (FL) is a distributed machine learning approach that enables clients to collaboratively train machine learning models using their local data, without the need to exchange raw data [3]. Instead, by sharing model parameters or intermediate results via a central server, data from different clients can be virtually fused and aligned, enabling clients to collaborate and learn from each other. Importantly, FL strikes a balance between data privacy and data sharing, embodying the principle that while "data remains unseen, it is still accessible" and "data stays stationary, but models are exchanged".

While Federated Learning (FL) offers potential, its client collaboration often falls short in performance due to the heterogeneous alignment of data domains across clients also known as Non-IID data. Recognizing the needs of the clients, previous studies [4], [5], [6], [7] have investigated the concept of personalized collaboration. Leading methods like FedFomo [5] and FedAMP [6] promote collaboration between client pairs with similar local models. Precisely, FedFomo gauges similarity through loss metrics, while FedAMP utilizes model parameter similarity. These methods operate under the assumption that clients with analogous models share high relevance and should therefore collaborate to enhance performance. However, our experiments indicate that neither loss nor model similarity conclusively indicates domain relevance among clients.

Motivation: We rethink the problem of personalized client collaboration in FL by focusing on measuring domain relevance

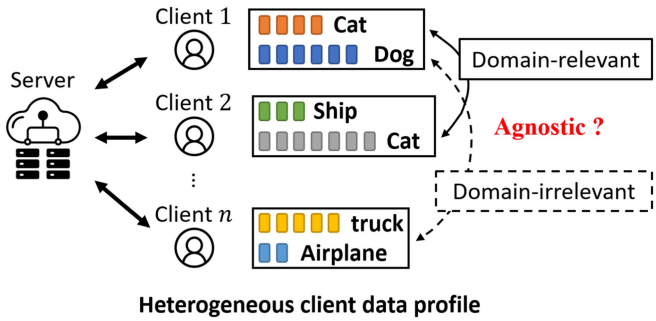


Fig. 1. Heterogeneous client data domain profiles in an agnostic federated learning system. client 1 and 2 are domain-relevant since they both have '*cat*', while client 1 and n are domain-irrelevant with no label overlap. But these domain relevances are agnostic to clients with the inherent FL privacy protection regulations.

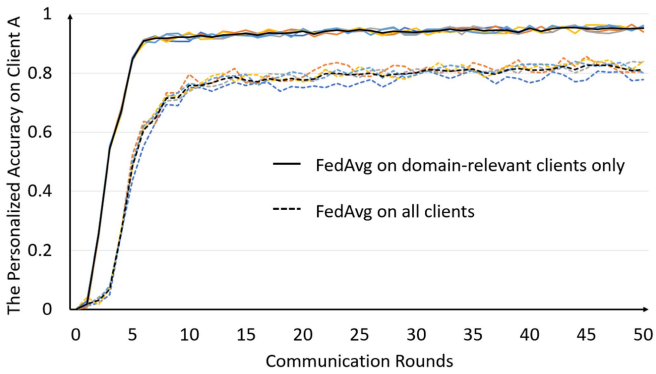


Fig. 2. The influence of domain relevance on the personalized performance of client A (MNIST). We repeat experiments for 5 times (indicated by different colors) and the black line is their average.

between clients. To elucidate our motivation, consider the example depicted in Fig. 1. While the '*cat*' on client 1 and 2 is domain-relevant in the data domain, the data domains between client 1 and client n are entirely unrelated. A core insight from our work is that collaboration between domain-relevant clients boosts performance, whereas involving unrelated clients can severely degrade outcomes.

To further certify our above key insight, we conducted a preliminary experiment in Fig. 2 by using the standard FedAvg Algorithm on MNIST with the following settings. We configured a setup with a total of 5 clients: A, B, C, D, E, assuming that the personalized task of client A is the even number classification, *i.e.*, $\{0, 2, 4, 6, 8\}$. The label distribution of other clients are: $B : \{0, 2, 4\}$, $C : \{6, 8\}$, $D : \{1, 3, 5\}$ and $E : \{7, 9\}$. It is very clear that class labels owned by client B and C overlap with client A. Thus, they ($B\&C$) are A's domain-relevant clients, while the other two clients ($D\&E$) are domain-irrelevant. Subsequently, we devised a personalized model for client A using the FedAvg algorithm under two distinct scenarios: In scenario (a), we aggregate the models of all 5 clients to generate a personalized model for client A. This scenario encompassed collaborations that intermingled with domain-irrelevant clients. In scenario (b), we only aggregate the models from client A, B, and C to generate a personalized model for client A, concentrating exclusively on collaboration with domain-relevant counterparts. We can observe that the personalized accuracy of user A converges rapidly within a few communication rounds when the collaboration is strictly with domain-relevant clients. Conversely, including domain-irrelevant collaborators in the mix degrades the final personalized accuracy of user A.

Given the privacy protection requirement in the FL system, it's impossible to directly conduct domain relevance analysis between clients on the data level, where the only available medium for information exchange is the model of each client. Therefore, different from the previous simple model similarity perspective, this paper introduces coalition game theory [8] to perform complex analysis on the model level so that the potential domain relevance at the data level can be accurately reflected. In this way, we can guarantee the domain relevance identification, while strictly adhering to the privacy protection requirement of FL. This theory aids each client in assessing the marginal contributions made by other clients' models to their own personalization process. The calculation of the average marginal contribution of a participating client's model considers all potential combinations of clients within the ongoing personalized coalition. This computation, also referred to as the Shapley Value (SV), encapsulates the collaborative impact of each client's model.

Expanding on this groundwork, we enhance the involvement of individual clients in coalitions and present a personalized FL aggregation algorithm. This algorithm repurposes the SV as aggregation weights, effectively steering the FL training procedure. Notably, this approach showcases robustness even in scenarios with highly Non-IID data distributions. We embark on a theoretical analysis of the convergence and generalization bounds of the proposed algorithm. Additionally, we notice that the local SV evaluation on each client requires them to download the model of others, which raises issues about communication overhead and privacy. Thus, we further utilize a shared feature extractor to reduce communication overhead and differential privacy techniques to protect model privacy.

To the best of our knowledge, this is the first time that coalition game theory is used as a guiding principle for the personalized collaboration process within FL. In summary, the principal contributions of this paper are four-fold:

- We revisit the personalized client collaboration problem in FL from the perspective of domain relevance and model this problem as a coalition game.
- We employ the insights from coalition game theory, particularly the Shapley Value (SV), to aid each client in identifying domain-relevant collaborators. This is achieved by assessing the marginal contributions of other clients to their own personalized performance.
- The SV from domain relevance analysis can be reused as aggregation weights to steer the FL training process, which implements a personalized FL aggregation algorithm without any extra information. The convergence and generalization bounds of the algorithm are theoretically analyzed.

- We conduct extensive experiments to validate the performance of our proposed algorithm, pFedSV, on datasets with different non-IID settings. The results show that pFedSV outperforms state-of-the-art baselines.

II. Related Work

Personalized Federated Learning. Recently, to address the client data heterogeneity challenge, Personalized Federated Learning (PFL) is proposed by utilizing the knowledge from other clients to customize a unique model for themselves, rather than using the traditional FL method to generate a single global model for all, which can significantly improve the model performance for every client in FL system. A series of surveys regarding the concept of "PFL" are proposed to summarize and generalize the key challenges and techniques in this field [9], [10], [11]. Following their novel insights, we organize the development vein of PFL in recent years as follows.

Initially, an additional fine-tuning step for the global model on each client's local dataset is a natural strategy for personalization [12], [13], which enables the global model to fit local data domains. Besides, some previous studies also attempted to enhance the robustness of global model under severe data non-IID level. They tried to add regularization term [14] or proximal term [15] to constraint the update of global model, which keeps the robust to all heterogeneous clients However, their methods are all based on the adjustment of a single global model scheme, which cannot satisfy the personalized demand of individual clients at the local data level, as the target distribution of clients in severe data Non-IID setting can be fairly different from the global average aggregation [16]. Therefore, a part of work, such as pFedHN, considers directly generating personalized parameters for each client's model [4]. While most other works try to promote collaboration between different clients to achieve mutual progress, FedFomo [5] and FedAMP [6] follow a similar idea that encourages pairwise collaboration among clients with similar model features, where the former uses loss similarity and the latter adopts parameter similarity. Clients who have higher similarity in these model features will be assigned higher aggregation weights, rather than the previous average. Moreover, considering the relevance of different clients, the cluster-based PFL techniques are proposed to group users with similar model features into a common subset, e.g., FedEM [17] and IFCA [18]. The basic idea of IFCA is to alternate between estimating the cluster identities and minimizing the loss functions based on the uploaded local models. FedEM is based on a flexible assumption that each local data distribution is a mixture of unknown underlying distributions. However, the cluster-based methods are limited since no knowledge is exchanged across clusters, where the extreme case is that each client forms a cluster with itself only.

Although the above collaboration methods have achieved good results, they still do not capture the essence of PFL: 1) Each client wants to collaborate with others who are truly relevant at the local data level, not model similarity. 2) Model aggregation is a multiwise process. Only considering pairwise relationships ignores the intertwined interactions among models. Thus, We introduce SV from coalition game theory to help each client accurately identify their domain-relevant collaborators with privacy guarantee, by complex marginal contribution analysis. Furthermore, the SV can also be reused as personalized model aggregation weights for each client.

Shapley Value for Federated Learning: The conventional FL framework is a multi-party architecture where clients collaboratively train a shared global model with data privacy protection. Considering the heterogeneity of clients in terms of data domain, hardware, resources, etc., the contribution of different clients to the single shared global model varies significantly, which is also very difficult to precisely quantify them. As a fair contribution evaluation metric, the Shapley Value from the cooperation game theory [19] can successfully solve this problem by measuring the marginal contribution of collaborators on the final outcome, where its calculation process considers the final results under various different combinations of collaborators.

Therefore, it's widely applied in various multi-party collaboration scenarios, such as FL. Wang et al. use SV in FL for various applications: 1) they measure the contribution of different clients for fair credit allocation [20], and 2) they quantify the importance of different features to the final prediction [21]. Song et al. achieve a fair profit allocation for clients in FL by using SV as the contribution index [22]. Furthermore, Yu et al. also utilizes the fair property of SV to design an incentive mechanism in FL [23].

However, they mainly utilize the desirable properties of SV to ensure the fairness of their contribution evaluation on different clients, but ignore that SV as a meaningful quantitative metric, can also guide the training process of FL. Some other works also notice that SV can be a effective guidance for typical FL training. Nagalapatti et al. propose to use SV-based model aggregation on heterogeneous client models to obtain a global model with higher performance [24]. Sun et al. present an adaptive SV-based weighting mechanism for the robustness of FL [25].

However, these works cannot be well generalized to the PFL scenario, since their server's general dataset can only enable global SV evaluation. The personalized SV evaluation requires local client data as metric, which is a significant challenge under FL data protection principle. In our work, the SV evaluation of the final model performance can help analyze the underlying data quality of different clients, without disclosing any data privacy. Besides, SV can also be reused as personalized aggregation weight to enhance model robustness against Non-IID data distribution.

III. The Essence of PFL Problem

A. Problem Formulation

Before the problem analysis, we first introduce the objectives of PFL and the corresponding problem formulation [1], [26]. PFL aims to customize personalized models for each client to accommodate their private data distribution through collaboration between a set of clients. Considering n clients $C_1, C_2, \ldots, C_n$ with the same structure of model $\mathcal{M}$ but parameterized by different weights $\theta_1, \theta_2, \ldots, \theta_n$, their respective personalized models can be denoted by $\mathcal{M}(\theta_i)$. Unlike traditional federated

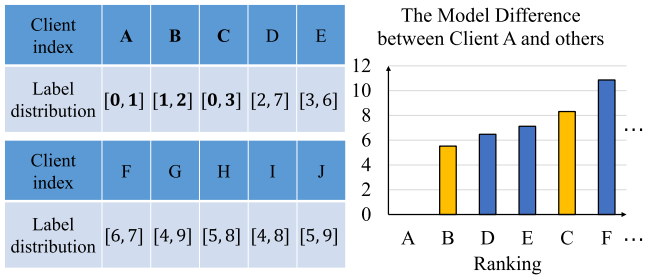


Fig. 3. The validation of model similarity theory for domain relevance identification, where the table shows the ground truth of client label distribution and the bar chart shows the model difference $||\theta_A - \theta_i||^2$ between client A and other client $i \in \{N\}$.

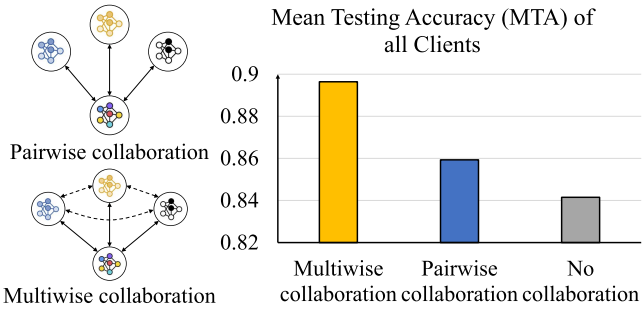


Fig. 4. The schematic of Multiwise vs. Pairwise collaboration and the experiment results on CIFAR-10 dataset with the pathological Non-IID setting.

learning, the private dataset $\mathcal{D}_i$ of each client i is uniformly sampled from their own distinct data distribution $\mathcal{P}_i$. Let ℓ_i denote the corresponding loss function for client i, and $\mathcal{L}_i$ the average loss over the private dataset $\mathcal{D}_i$ is denoted by $\mathcal{L}_i(\theta_i) = \frac{1}{d_i}\sum_{j\in\mathcal{D}_i} \ell_i(j, \theta_i)$, where d_i is the data size of $\mathcal{D}_i$ and j is one of the data samples in $\mathcal{D}_i$. The optimization objective of PFL is

$$\Theta^* = \arg\min_{\Theta} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}_i(\theta_i), \tag{1}$$

where Θ is the set of personalized model parameter $\{\theta_i\}_{i=1}^n$. Next, we will delve into the root causes of the problems through various pre-experiments analysis and present our multiwise collaboration solution: Shapley value, to address these problems.

B. Root Causes of PFL Problems

Domain Relevance: According to extensive previous work for data Non-IID problem in FL [15], [26], [27], [28], the model performance degradation is due to the client data domain heterogeneity. However, the inherent data privacy protection of FL makes it difficult to identify other domain-relevant clients, when facing an agnostic system. Since the client models are the only communication intermediary in this situation, previous work directly adopts one-to-one model similarity test to represent domain relevance, i.e., clients with higher model similarity will be regarded as having higher domain relevance. But, there are some flaws lurking behind this theory, which can cause wrong identification. In the table of Fig. 3, we show the ground truth of all client label distribution, where the data distribution is pathological Non-IID partition on CIFAR-10 dataset, and numbers $0 \sim 9$ represent the index of different labels. Take client A with labels [0,1] as an example, client B with labels [1,2] and client C with labels [0,3] are its domain-relevant clients, since them both have overlap label of A. We use euclidean distance, i.e., $||\theta_A - \theta_i||^2, i \in \{N\}$, to measure the model difference between client A and other clients in Fig. 3. If the theory is true, the model differences of B and C should be the smallest among all clients, i.e., $||\theta_A - \theta_B||^2 \approx ||\theta_A - \theta_C||^2 < ||\theta_A - \theta_i||^2, i \in N \setminus \{B, C\}$, while the results in Fig. 3 are not consistent with it.

Multiwise Collaboration Weights: Another key-point is the personalized model aggregation within the coalition to generate client-specific model. The previous methods adopted pairwise collaboration by comparing model similarities one-to-one and assigning proportional aggregation weights based on their magnitudes, which is demonstrated in Fig. 4. However, imagine a scenario where the client's current model is a carriage, and every other client's model is a force that moves the carriage in a certain direction, and the destination of the carriage is the client's optimal personalized model. Obviously, the movement of carriage is the result of multiple forces combination, which indicates that the multiwise influences among collaborators must be considered when generating the personalized model aggregation weights. Under the same conditions that each client is informed in advance about respective domain-relevant clients, we conduct extensive experiments, where the only variable is the collaboration methods among clients when generating aggregated weights. The results in Fig. 4 indicating that multiwise collaboration outperforms pairwise collaboration.

C. Domain-Relevant Coalition Formation and Personalized Model Generation

1) Preliminaries of SV: Consider each client as a player in the coalition game, where $N = \{1, 2, \ldots, n\}$ denotes the set of players. A *utility function* $v(S) : 2^n \to \mathbb{R}$ assigns to every coalition $S \subseteq N$ a real number representing the gain obtained by the coalition as a whole. By convention, we assume that $v(\emptyset) = 0$. Formally, let $\pi \in \Pi(N)$ denote a permutation of clients in N, and $C_\pi(i) = \{j \in \pi : \pi(j) < \pi(i)\}$ is a coalition containing all predecessors of client i in π. The SV of client i is defined as the average marginal contribution to all possible coalitions $C_\pi(i)$ formed by other clients:

$$\varphi_i(v) = \frac{1}{|N|!}\sum_{\pi\in\Pi}[v(C_\pi(i) \cup \{i\}) - v(C_\pi(i))]. \tag{2}$$

The formula in (2) can also be rewritten as:

$$\varphi_i(v) = \sum_{S\subseteq N\setminus\{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!}[v(S \cup \{i\}) - v(S)]. \tag{3}$$

The SV has several desirable and unique properties, which can achieve domain-relevant coalition formation for each client and perform personalized model generation in the coalition.

2) SV for Domain Relevance:

- *Symmetry:* Two clients who have the same contribution to the coalition should have the same value. That is, if client i and j are equivalent in the sense of $v(S \cup \{i\}) = v(S \cup \{j\}), \forall S \subseteq N \setminus \{i, j\}$, then $\varphi_i = \varphi_j$.
- *Null Player:* Client with zero marginal contributions to all possible coalitions is null player and receive zero payoff, i.e., $\varphi_i = 0$ if $v(S \cup \{i\}) = 0$ for all $S \subseteq N \setminus \{i\}$.

The *Symmetry* and *Null Player* properties in SV can assist each client in precisely identifying their own domain-relevant clients, where those irrelevant clients will be identified as *null Player*.

The workflow of domain-relevant client identification is introduced below: in each communication round t, each client i will first upload their local updated model θ_i^t to the server, forming a model pool $\{\theta_i^t\}_{i=1}^n$ on the server-side. Next, they will download other clients' models from the model pool to construct their own domain-relevant coalition, and then perform personalized model aggregation. However, in the agnostic federated learning system, not all clients are available at the beginning, which means the members of domain-relevant coalition is dynamical reconstructed during the training.

First, we construct a model download vector for each client based on the relevance score, that is, for each client $i, i \in N$, it generates an n-dimensional relevance vector $\phi^{i,t} = [\phi_1^{i,t}, \ldots, \phi_n^{i,t}]$, where $\phi_j^{i,t}$ denotes the relevance score of client j to i in t-th round and we have $\phi^{i,t=0} = \vec{0}$. We choose to download the models of those clients with top-k relevance score in the vector (Note: they randomly download k other clients' models in the first round since it's initialized as a all-zero vector). Then, each client can form a personalized coalition set $S_{i,k}^t$, which contains the client indexes of its own and those downloaded. Now, client i can perform SV evaluation in its personalized coalition $S_{i,k}^t$, by using the following coalition game and the local validation dataset $\mathcal{D}_{V_i}$. We define a coalition game $(\{\theta_j^t\}_{j \in S_{i,k}^t}, v)$, where v is a utility function that assigns a value to each client subset $X \subseteq S_{i,k}^t$. Here, we define the value as the performance $\mathcal{A}$ of the model θ_X^t generated from X on the validation dataset $\mathcal{D}_{V_i}$ as follows.

$$\theta_X^t = \frac{1}{|X|} \sum_{j \in X} \theta_j^t, \text{and } v(X, \mathcal{D}_{V_i}) = \mathcal{A}(\theta_X^t, \mathcal{D}_{V_i}). \quad (4)$$

Then, we can obtain the SV $\varphi_j^t, j \in S_{i,k}^t$ of all clients in the personalized coalition $S_{i,k}^t$ from the coalition game $(\{\theta_j^t\}_{j \in S_{i,k}^t}, v)$ in t-th round according to (2). Next, client i updates the relevance score of its relevance vector to $\phi^{i,t+1}$ as below:

$$\phi_j^{i,t+1} = \alpha \phi_j^{i,t} + (1 - \alpha)\varphi_j^t, \forall j \in S_{i,k}^t. \quad (5)$$

Intuitively, a larger relevance score of client j means that it contributes more to the personalized performance of client i, and thus has a higher likelihood of being its domain-relevant client. Besides, we notice that the relevance vector is unstable in the initial few rounds because not all models of domain-relevant clients can be downloaded. It requires several rounds of iterative updates to screen them with the top-k scheme as below. By definition, when other clients' models negatively affect the personalized performance in the coalition game, its SV can be negative, so the irrelevant clients' scores will rapidly decrease to negative in the iterations and thus be excluded. For example, if client i has 2 domain-relevant clients in total 20 clients and it downloads top-5 other clients' models per round, then it takes at most 5 rounds to identify all domain-relevant clients. The convergence analysis of Dynamic Top-k Download Mechanism is elaborated later in Section IV-B.

Dynamic top-k download mechanism: to reduce communication overhead, the download number k in each round can be dynamically adjusted. With iterative updates, only domain-relevant clients can remain positive relevance score, so when the number of clients with positive score does not match the current value k, we dynamically adjust it to ensure that all downloads are for the necessary domain-relevant clients.

3) SV for Multiwise Collaboration Weights:

- *Group Rationality:* The gain of the entire coalition S is completely distributed among all clients in S, i.e., $v(S) = \sum_{i \in S} \varphi_i$.
- *Linearity:* The values under multiple utilities sum up to the value under a utility that is the sum of all these utilities: $\varphi_i(v) + \varphi_i(u) = \varphi_i(v + u)$. Also, for every $i \in N$ and any real number a, it has $\varphi_i(av) = a\varphi_i(v)$.

The *Group Rationality* and *Linearity* properties perfectly fit the demand of personalized model aggregation with multiwise collaboration in the coalition. According to (2), the computation of SV requires exploring extensive permutations among clients within the coalition game, hence the process naturally considers the complex multiwise influences on the final results. Furthermore, the *Group Rationality* property guarantees that the target of all clients within the coalition is the same, i.e., to achieve the best performance for the current client i, which also means the optimal personalized model parameters. And the *Linearity* property naturally fits into the model aggregation process, i.e., the improvement of personalized accuracy by aggregating other client models into their own is fully reflected in the SV of the model, where a larger positive SV indicates a larger positive contribution to performance improvement and vise verse.

Based on the SV φ_j^t of all clients $j \in S_{i,k}^t$ in (5), the downloaded models are assigned a real number that represents their marginal contribution to the personalization of the current client i, where a positive number indicates a positive effect and vice versa. Therefore, we first need to exclude those models of irrelevant clients with negative SV out of the multiwise collaboration in current round, and only compute the weights for remaining domain-relevant clients within the coalition as follows:

$$w_j^t = \frac{\max(\varphi_j^t, 0)}{\|\theta_i^t - \theta_j^t\|}, \quad (6)$$

where we adopt the model differences $\|\theta_i^t - \theta_j^t\|$ to further fine-tune the resulting weights. Then, we perform 0-1 normalization on the previous weights to obtain their personalized aggregation weights $w_j^{t*} = \frac{w_j^t}{\sum_j w_j^t}$, which maintains $w_j^{t*} \in [0, 1]$

and $\sum_j w_j^{t*} = 1$. Finally, we generate the personalized model of client i in t-th round based on the following multiwise collaboration:

$$\theta_i^{t*} = \sum_j w_j^{t*} \theta_j^t, \quad \forall j \in S_{i,k}^t. \tag{7}$$

Note that we perform SV evaluations in each round to accommodate small changes in multiwise influences due to parameter changes after client local model training.

IV. The pFedSV Algorithm

Based on the above solution frame, we propose our pFedSV Algorithm, where the whole workflow is demonstrated in Algorithms 1 and 2. In the beginning, each client initialize their model parameters θ_i and the relevance vector ϕ^i (Line 1-2). Then in each round t, they update the model parameters to θ_i^t by E local epochs training and upload them to the server (Line 5). Next, they download k copies of other clients' model parameters according to the dynamic top-k download mechanism (Line 6). At this point, the basic conditions of each client's coalition game for their own model personalization are available. First, they form a coalition game $(\{\theta_j^t\}_{j \in S_{i,k}^t}, v)$, where $S_{i,k}^t$ is the model parameter set consisting of k downloaded model parameters and their own (Line 7). Then, the SV evaluation process is performed to obtain the SV of each model parameters in $S_{i,k}^t$ (Line 8), which will be elaborated in Algorithm 2 later. Next, the obtained SV are used to address two challenges: updating the relevance vector of each client for identifying their domain-relevant clients (Line 9), and calculating the multiwise aggregation weights for model personalization (Line 10). Finally, each client performs the respective weighted aggregation to obtain new model parameters as the starting point for the next round $t+1$. The current Algorithm 1 is based on the original whole model downloading, the further version of Algorithm with global feature extractor will be provided in the Appendix, available online.

Since the time complexity required to accurately evaluate SV is exponential to the number of players, we need an approximation algorithm to make the trade-off. According to (2), the calculation of SV can be viewed as an expectation calculation problem, so we adopt a widely accepted Monte Carlo sampling technique to approximate the SV [29], [30], [31]. The related details are elaborated in Algorithm 2. First, we randomly sample R permutations of $S_{i,k}^t$ out of total $|S_{i,k}^t|!$ permutations to form a set P (Line 1). Then, for each permutation, we scan it from the first element to the last and calculate the marginal contribution for every newly added model parameters (Line 3-5). Perform the same procedure for all R permutations and the approximation of SV is the average of R calculated marginal contributions (Line 6). As the number of samples R gradually increases, Monte Carlo sampling will eventually be an unbiased estimate of the SV.

A. Convergence of SV Evaluation Approximation.

The computation complexity for precise SV evaluation is exponential to the number of players. According to (2), the computation process can be viewed as an expectation calculation problem, thus the Monte Carlo sampling technique can be used to approximate the SV. It will converge to an unbiased estimate of the SV as the increasing of sampling number R. It's proofed that $R = 3|S_{i,k}^t| \ll |S_{i,k}^t|!$ Monte Carlo sampling number is sufficient for convergence, with a small approximation bound $\epsilon > 0$ [31].

Algorithm 1: Shapley Value Based Personalized Federated Learning on Whole Model (pFedSV).

Input: n, N, $\{\theta_i\}_{i=1}^n$, k, E, T, R and $\mathcal{D}_{V_i}$.
Output: $\{\theta_i^*\}_{i=1}^n$: clients' personalized model parameters.
1: Initialize the clients' model parameters $\{\theta_i\}_{i=1}^n$. 2
2: Initialize clients' relevence vector: $\phi^{i,t=1} = \vec{0}$, $\forall i \in N$.
3: **for** round $t = 1, 2, \ldots, T$ **do**
4: **for** client $i = 1, 2, \ldots, n$ **do**
5: update its model parameter to θ_i^t via E local epochs and upload to the server.
6: download k copies of other clients' model parameters from server with the dynamic top-k download mechanism.
7: $S_{i,k}^t \leftarrow \theta_i^t \cup \{k$ downloaded model parameters$\}$.
8: $\varphi_j^t \Leftarrow$ **SV_evaluation**$(S_{i,k}^t, \mathcal{D}_{V_i}, R), \forall j \in S_{i,k}^t$. {Details in Algorithm 2}
9: $\phi_j^{i,t+1} = \alpha \phi_j^{i,t} + (1-\alpha)\varphi_j^t, \forall j \in S_{i,k}^t$
10: $w_j^{t*} = \frac{w_j^t}{\sum_j w_j^t} \Leftarrow w_j^t = \frac{\max(\varphi_j^t, 0)}{\|\theta_i^t - \theta_j^t\|}, \forall j \in S_{i,k}^t$.
11: $\theta_i^{t*} = \sum_j w_j^{t*} \theta_j^t, \forall j \in S_{i,k}^t$.
12: **end for**
13: **end for**

Algorithm 2: Shapley Value Evaluation.

Input: $S_{i,k}^t$, $\mathcal{D}_{V_i}$, R.
Output: $\varphi_j^t, \forall j \in S_{i,k}^t$.
1: $P \leftarrow$ set of R permutations of $S_{i,k}^t$.
2: **for** client $j \in S_{i,k}^t$ **do**
3: **for** permutation $p \in P$ **do**
4: $X_{p,j}^t = \{l | l \in S_{i,k}^t \wedge p(l) \leq j\}$;
5: $a_j^p \leftarrow v(\{X_{p,j}^t \cup j\}, \mathcal{D}_{V_i}) - v(X_{p,j}^t, \mathcal{D}_{V_i})$
6: $\varphi_j^t \leftarrow \varphi_j^t + \frac{1}{|P|} a_j^p$.
7: **end for**
8: **end for**

B. Convergence Analysis of Dynamic Top-K Download Mechanism

Assume that there are total n clients with 100% participation, the local data distributions of these clients follow the pathological data Non-IID setting, where each client is randomly assigned m types of labels. An example of client label distribution on the CIFAR-10 dataset with $m = 2$ is shown in Fig. 3 for reference. Domain heterogeneity is defined as each client's label distribution is different, while domain relevance is defined as there are same class labels between different clients. Therefore, we can observe from the ground truth of Fig. 3 that each client has m

other domain-relevant clients in this setting from an omniscient perspective.

Suppose that the initial model download number for each client is k. Then, we provide the convergence proof of our dynamic top-k download mechanism. Take the personalization process of client A as an example, there are two conditions for the settings of hyperparameters m and k ($m < k$ or $m > k$), and we will explain them one by one.

When $m < k$: In the first round, each client will randomly download k copies of other clients' models from the server-side and there are various (C_n^k) possible model combinations.

- *For the best case*, other m domain-related clients' models are all included in the initial k copies, that is for $\forall i \in \{m\}$, we have $i \in \{k\}$. Thus, we can identify all domain-relevant clients of client A in the first round, where the SV of the domain-relevant clients is positive and the domain-irrelevant clients are negative.
- *For the worst case*, none of the m domain-related models is included in the first k copies, that is for $\forall i \in \{m\}$, we have $i \notin \{k\}$. Next, we proof the maximum number of rounds that is required to identify the m domain-relevant clients from all n clients when the worst case occurs in each round. For the first round, since k copies of models are all from the domain-irrelevant clients, their SV will be negative in the evaluation process, which makes their relevance score be negative after updating. Therefore, according to the top-k rule, these clients will not be selected in the next round because the relevant scores of other clients who have never been selected are the initial 0, which is larger than negative scores. The worst case will continue until a certain round t, which satisfies $tk > n - m - 1$ (1 is client A itself). It means that in round t, we have excluded all domain-irrelevant clients with negative SV, and the remaining clients are all domain-relevant clients. Since $k > m$ (they are both integers), we have $(t+1)k = tk + k > n - m - 1 + k > n + (k - m - 1) \geq n$, which means that we must be able to find all domain-relevant clients in the next round $t+1$. Finally, we prove that it takes at most $\lceil \frac{n-m-1}{k} \rceil + 1$ round to identify all other domain-relevant clients.

When $m > k$, following the similar logic as above, we can get the subsequent convergence proof.

- *For the best case*, since $m > k$, we cannot include all m domain-relevant clients in the first round with only k downloaded models. Therefore, the process will continue until all clients are scanned by once. Thus, we need $\lceil \frac{m}{k} \rceil$ round to identify all domain-relevant clients.
- *For the worst case*, we need $\lceil \frac{n-m-1}{k} \rceil$ rounds to exclude all domain-irrelevant clients and then we still need up to $\lceil \frac{m}{k} \rceil$ rounds to identify all domain-relevant clients. Finally, it takes at most $\lceil \frac{n-m-1}{k} \rceil + \lceil \frac{m}{k} \rceil$ rounds.

Normally, to ensure efficient traversal, we will set a large value of k at the beginning. Although a large k leads to a large communication overhead in the beginning, it can help the client rapidly scan all other clients and converge to a specific value $k = m$, which is equal to the number of other domain-relevant clients.

C. Convergence Analysis of pFedSV.

We proof that pFedSV can assist each client converge to their respective local optimums under the following assumptions: 1) $\mathcal{L}_1, \ldots, \mathcal{L}_n$ are all μ-strongly-convex, 2) $\mathcal{L}_1, \ldots, \mathcal{L}_n$ are all L-smooth, 3) the variance of stochastic gradients in each client is bounded by σ_i^2 and 4) the expected squared norm of stochastic gradients is uniformly bounded by G^2.

Theorem 1: Let all above assumptions hold and μ, L, σ_i, G are defined therein. Choose $\kappa = \frac{L}{\mu}$, $\gamma = \max\{8\kappa, E\}$ and the learning rate $\eta_t = \frac{2}{\mu(\gamma+t)}$. Then, each client in pFedSV satisfies

$$\mathbb{E}[\mathcal{L}_i(\theta_i)] - \mathcal{L}_i^* \leq \frac{\kappa}{\gamma + T - 1}\left(\frac{2B}{\mu} + \frac{\mu\gamma}{2}\mathbb{E}\|\theta_i^1 - \theta_i^*\|^2\right) \quad (8)$$

The full version about the convergence analysis of pFedSV algorithm will be elaborated in the Appendix, available online.

D. Generalization Bounds of pFedSV.

In this section, We theoretically prove that the performance of pFedSV can outperform conventional FedAvg algorithm and local training only, through the theorem from domain adaptation [32].

Theorem 2: For each client $i \in N$, we denote its local distribution and empirical distribution as $\mathcal{D}_i$ and $\hat{\mathcal{D}}_i$. The model parameters learned on $\hat{\mathcal{D}}_i$ is denoted by $\theta_{\hat{\mathcal{D}}_i}$. Then we have

$$\begin{aligned}\mathcal{L}_{\mathcal{D}_i}\left(\sum_j w_{\hat{\mathcal{D}}_j}^{t*}\theta_{\hat{\mathcal{D}}_j}^t\right) &\leq \mathcal{L}_{\mathcal{D}_i}\left(\frac{1}{n}\sum_j \theta_{\hat{\mathcal{D}}_j}^t\right) \\ &\leq \mathcal{L}_{\hat{\mathcal{D}}_i}(\theta_{\hat{\mathcal{D}}_i}) + \frac{1}{n}\sum_j\left(\frac{1}{2}d(\mathcal{D}_i, \mathcal{D}_j) + \xi_j\right) + \sqrt{\frac{\log\frac{2n}{\delta}}{2\,m}} \quad (9)\end{aligned}$$

where $w_{\hat{\mathcal{D}}_j}^{t*}$ is the SV-based aggregation weight, $d(\cdot)$ measures the distribution discrepancy between two distributions, m is the number of samples per local distribution and ξ_j is the minimum of the combined loss $\mathcal{L}_{\hat{\mathcal{D}}_i} + \mathcal{L}_{\hat{\mathcal{D}}_j}$. The details of generalization bounds are elaborated in the Appendix, available online.

V. Communication Overhead Reduction & Model Privacy Protection

Since each user needs to download many model copies of other users to perform their local SV evaluation in our pFedSV algorithm, we are also aware of the potential communication overhead increase and model privacy issues arising from this model downloading process, and provide solid solutions to address them in this section.

A. Communication Overhead Reduction

Except for the top-k dynamic mechanism in Section III-C-2, we further exploit the advantage of a global shared feature extractor between users to reduce the communication overhead. Specifically, for different learning tasks (i.e., image classification and next word prediction), the model can be divided into two parts: feature extractor and classifier, where the former has a generic function for all users, and the latter is unique for

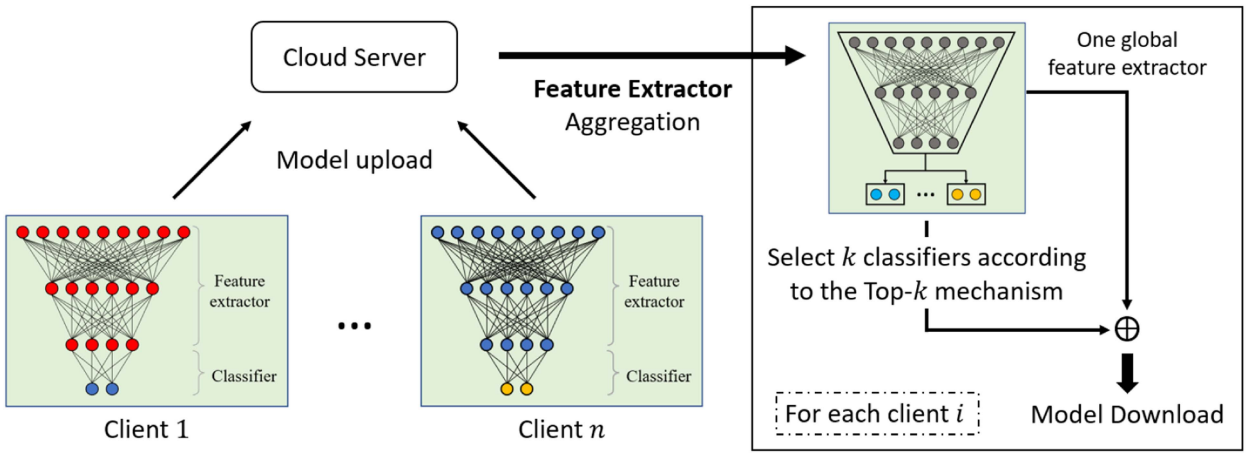


Fig. 5. The modified federated learning workflow is based on model splitting, where the model of each user is divided into a feature extractor and classifier. The server only generates a global shared feature extractor among different users while maintaining their personalized classifiers.

different user's local data domains [7]. According to the latest research [33], they measure the Centered Kernel Alignment (CKA) similarity between the representations from the same layer of different clients' local models, on standard CNN [34]. The observation is clear: comparing different layers in the local models learned on different clients, the similarity of feature extractors among different client local models is very high, while the classifiers have the lowest similarity

Therefore, for the personalization of each user, the most important thing they need to focus on is the classifier of other users, while the feature extractor part can be shared. Following this insight, each user only needs to download one global shared feature extractor and several classifiers of other users to reduce the communication overhead, not the whole model before. And the whole model of other users can be reconstructed by replacing different classifiers. The modified federated learning workflow is demonstrated in Fig. 5.

Nevertheless, since our pFedSV is a general personalization algorithm for different learning tasks, other classical techniques, such as model quantification and compression, can also be applied for further communication reduction.

B. Model Privacy Protection

For the issue of model privacy, we have achieved anonymity by removing any information related to the client's identity from the downloaded models during the entire process of pFedSV. Since the model itself may still imply client private data information in these parameters, we design a more effective privacy protection method, by adopting the (ϵ, δ)-differential privacy (DP) to address the privacy issue in our scenario [35]. We add Gaussian noise into the model parameters after client's local training process, which can guarantee the model with DP.

In brief, DP ensures that, given two nearly identical datasets, querying one dataset produces results with nearly the same probability as querying the other, which is under the control of δ and ϵ. In particular, the DP in our scenario can reduce the connection between the local dataset and the trained model parameters. More noise makes the model more private at the cost of performance and we conduct extensive experiments to illustrate whether pFedSV can retain its performance with more privacy protection (add more noise). The experiment results in the later section indicate that, under an appropriate DP noise ($\delta = 1$), the performance of our pFedSV algorithm with DP-based Noise Addition can still outperform all other personalized baselines.

VI. Experiments

A. Experimental Setup

In this section, we will show all the experiment setups, including hyperparameter settings, datasets, baselines, etc.

1) Dataset, Model & Machine Configurations: Based on prior work [36], [37], we conduct our experiments on the following datasets: MNIST [34], Fashion-MNIST (F-MNIST) [38], CIFAR-10 [39], and CIFAR-100. For the model structure on different datasets, We use the same CNN architecture as in [36]. All our experiments are run on the following machine configurations: CPU (i9-10900 K) and GPU (one RTX 3090).

2) Baselines & Evaluation Metric: We evaluate the performance of pFedSV by comparing it with the state-of-the-art PFL algorithms, including pFedMe [14], pFedHN [4], FedFomo [5] and FedAMP [6]. For a more comprehensive understanding, we also compare with the classical single global model methods FedAvg [3], FedAvg with fine-tuning (FedAvg+FT) and FedProx [15], as well as the simplest separate local training named *separate*, where each client individually train their own model without collaboration. The performance of all algorithms is evaluated by the mean testing accuracy (MTA), which is the average of the testing accuracy on all clients, and the $\pm$ indicates the error range of the MTA after 5 repeated experiments.

3) Non-IID Data Setting.: For each used dataset, we adopt two different Non-IID data settings as follows: • The pathological Non-IID data setting: each client is randomly assigned two types of labels and the privacy data is not similar between any two clients, which is shown as the Table in Fig. 4. (20 types of

TABLE I
THE MTA WITH THE PATHOLOGICAL NON-IID DATA SETTING

Methods	MNIST		FMNIST		CIFAR-10		CIFAR-100	
	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients
Seperate	96.11 ± 0.28	93.27 ± 3.68	92.35 ± 0.43	91.42 ± 2.69	84.15 ± 2.13	75.57 ± 4.08	73.57 ± 5.13	68.57 ± 4.25
FedAvg	91.74 ± 1.68	78.46 ± 1.14	90.31 ± 2.49	75.63 ± 4.73	57.67 ± 4.16	44.64 ± 4.75	50.57 ± 3.71	43.16 ± 4.68
FedProx	90.12 ± 0.73	78.45 ± 1.83	90.16 ± 3.05	78.83 ± 3.49	55.68 ± 2.67	45.75 ± 4.39	49.21 ± 3.69	41.08 ± 5.27
IFCA	92.86 ± 1.57	86.73 ± 2.05	90.01 ± 2.38	82.63 ± 3.59	71.69 ± 3.25	60.23 ± 3.94	61.37 ± 4.16	52.44 ± 4.68
FedEM	95.05 ± 1.28	91.53 ± 1.87	92.27 ± 2.68	87.61 ± 4.07	82.38 ± 3.56	71.42 ± 5.07	72.39 ± 5.76	62.84 ± 4.28
FedAvg+FT	94.38 ± 1.06	90.51 ± 1.67	91.18 ± 3.54	89.49 ± 4.51	81.34 ± 3.24	70.13 ± 5.68	71.08 ± 5.14	64.38 ± 4.69
pFedMe	93.75 ± 1.34	86.57 ± 2.61	92.46 ± 1.72	85.39 ± 2.97	80.48 ± 4.59	70.15 ± 5.86	71.56 ± 4.79	60.85 ± 5.26
FedFomo	96.90 ± 0.87	93.71 ± 2.05	94.10 ± 0.65	92.78 ± 1.92	85.93 ± 3.02	74.36 ± 2.15	76.89 ± 3.54	69.21 ± 4.37
FedAMP	95.82 ± 1.37	92.59 ± 1.88	93.26 ± 2.14	91.46 ± 2.04	84.32 ± 3.69	72.91 ± 2.83	75.38 ± 3.19	67.02 ± 4.15
pFedHN	96.53 ± 0.84	94.16 ± 1.38	94.97 ± 0.86	93.69 ± 1.58	86.38 ± 2.72	76.62 ± 3.05	77.24 ± 3.86	70.58 ± 4.57
pFedSV(Ours)	$\mathbf{98.01 \pm 0.83}$	$\mathbf{96.94 \pm 1.75}$	$\mathbf{96.16 \pm 0.58}$	$\mathbf{94.68 \pm 2.36}$	$\mathbf{89.64 \pm 1.88}$	$\mathbf{80.65 \pm 3.78}$	$\mathbf{80.57 \pm 4.37}$	$\mathbf{72.61 \pm 4.73}$

10 clients with 100% and 100 clients with 10% participation in each round.
Where bold indicates the best result among all methods.

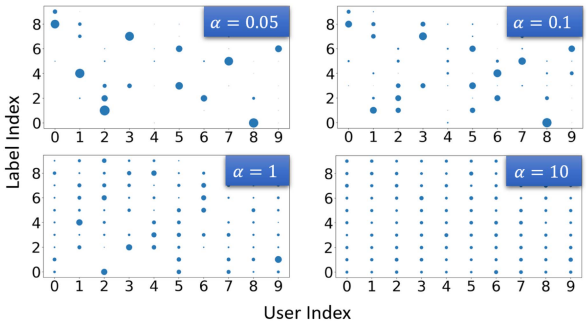


Fig. 6. The visualization of Dirichlet data Non-IID setting on MNIST, where x-axis indicates the client index, y-axis indicates the label index, and the size of scattered points indicates the number of training samples owned by the client.

labels per client on CIFAR-100 with 10 clients) • The Dirichlet Non-IID data setting **Dir**(α), which uses different α values to adjust the data Non-IID level. A small α means high data heterogeneity, that is, it makes the client label distribution more biased [40], and vice versa. To provide a clear understanding, the visualization of the Dirichlet setting with different α values on clients' data heterogeneity is shown as in Fig. 6.

4) Implementation Details.: We consider two FL scenarios with different client scales: total 10 clients with 100% participation and total 100 clients with 10% participation. We set the training parameters as 5 local epochs, the same number of communication rounds (20 rounds for the former and 100 rounds for the latter), and learning rates (0.01 for MNIST and FMNIST, 0.1 for CIFAR-10). For the SV related hyper-parameters, we set the Monte Carlo sampling number as $R = 3|S^t_{i,k}|$, where the number of model parameters downloaded for each round is $k = 5$ in the beginning. Note that k is dynamically adjusted according to the dynamic top-k download mechanism in *SV for domain relevance* of Section III-C. The open source will be available in GitHub after the acceptance (URL).

B. Performance Analysis

In this section, we will demonstrate the performance of our pFedSV compared to all the state-of-the-art benchmarks and analyze the experiment results in detail.

1) Results on the Different Non-IID Data Setting.: Table I demonstrate the MTA of all methods with the pathological Non-IID data setting. Since each client has only two types of labels, which significantly simplifies the complexity of the classification task for each client, the high performance of separate on all datasets reflects the simplicity. However, the pathological Non-IID data setting is a great challenge for the single global model methods, we can observe that FedAvg and FedProx suffer from significant performance degradation on all datasets, since its global aggregation will contain models of domain-irrelevant clients and thus lead to severe instability in the gradient optimization process [41]. For the other PFL methods, FedAvg+FT, pFedMe, FedFomo, FedAMP, pFedHN and our pFedSV all realize a promising performance on all datasets. FedAvg+FT takes several local fine-tuning steps to tune the poor global model back to adapt the local Non-IID data distribution. The pFedMe proposes novel regularized loss functions based on Moreau envelopes to decouple the personalized optimization from the global model learning. The pFedHN is more specific in that it directly generates personalized parameters for each client's model through another hypernetwork. The good performance of FedFomo and FedAMP are achieved by adaptively encourage more pairwise collaboration between clients with similar models to form their own personalized model. Our pFedSV can outperform all other baselines since it considers the multiwise influences among clients to help them identify their domain-relevant coalition and generate personalized aggregation weights with multiwise collaboration.

Table II illustrates the MTA of all methods on the Dirichlet Non-IID data setting ($\alpha = 0.1$). As we know from the visualization in Fig. 3, this setting (Dirichlet $\alpha = 0.1$) is much more challenge than pathological, which is reflected in the significant performance reduction of all methods. Nevertheless, our pFedSV is still guaranteed to outperform all other baselines. Please note that the low accuracy in Table II with 100 clients scale is due to only 10% participation in each round.

2) Relevance Score & Multiwise Collaboration Weights.: The superior performance of pFedSV on domain relevance identification comes from the various desirable properties of SV, Fig. 7 visualize the relevance vector ϕ^i of each client after

TABLE II
THE MTA WITH THE DIRICHLET NON-IID DATA SETTING ($\alpha = 0.1$) ON DIFFERENT DATASETS

Methods	MNIST		FMNIST		CIFAR-10		CIFAR-100	
	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients
Seperate	74.05 ± 2.11	59.81 ± 5.73	60.18 ± 6.42	58.22 ± 6.73	40.53 ± 7.20	36.15 ± 6.88	35.43 ± 3.87	30.05 ± 5.49
FedAvg	43.57 ± 3.75	30.15 ± 4.82	40.58 ± 4.16	36.49 ± 5.07	33.81 ± 5.07	26.82 ± 6.43	26.17 ± 4.27	20.33 ± 5.27
FedProx	47.49 ± 4.18	44.76 ± 5.49	43.09 ± 4.82	40.34 ± 4.72	35.76 ± 5.18	29.91 ± 5.58	29.62 ± 5.13	23.27 ± 4.69
IFCA	58.67 ± 2.69	54.58 ± 3.79	56.29 ± 4.16	51.04 ± 4.38	43.09 ± 4.88	40.67 ± 4.86	39.28 ± 4.11	31.89 ± 4.20
FedEM	66.53 ± 2.74	60.28 ± 4.05	61.79 ± 3.62	57.41 ± 4.28	51.09 ± 4.58	45.82 ± 5.07	41.39 ± 3.76	35.88 ± 4.61
FedAvg+FT	65.72 ± 3.84	55.57 ± 4.26	57.27 ± 4.13	52.83 ± 5.01	43.42 ± 5.29	40.05 ± 5.22	36.33 ± 3.86	32.55 ± 4.37
pFedMe	64.39 ± 4.08	58.02 ± 3.51	60.27 ± 3.59	56.81 ± 4.01	50.73 ± 4.29	44.21 ± 5.09	40.29 ± 3.57	34.94 ± 3.78
FedFomo	72.54 ± 2.18	63.07 ± 2.54	64.75 ± 3.42	60.49 ± 3.72	53.83 ± 4.57	48.35 ± 5.29	45.91 ± 3.06	37.51 ± 3.09
FedAMP	70.15 ± 3.02	60.28 ± 3.11	62.28 ± 2.53	58.94 ± 3.14	51.57 ± 4.03	46.05 ± 4.48	43.67 ± 3.55	36.40 ± 3.76
pFedHN	73.35 ± 2.04	62.57 ± 4.11	62.95 ± 3.44	59.55 ± 4.15	52.82 ± 3.88	47.19 ± 5.83	45.33 ± 3.45	37.38 ± 3.77
pFedSV(Ours)	$\mathbf{78.17 \pm 1.59}$	$\mathbf{70.76 \pm 2.41}$	$\mathbf{71.47 \pm 1.86}$	$\mathbf{66.63 \pm 2.03}$	$\mathbf{61.18 \pm 1.67}$	$\mathbf{56.76 \pm 1.85}$	$\mathbf{50.46 \pm 2.47}$	$\mathbf{42.25 \pm 3.13}$

10 clients with 100% and 100 clients with 10% participation in each round.
Where bold indicates the best result among all methods.

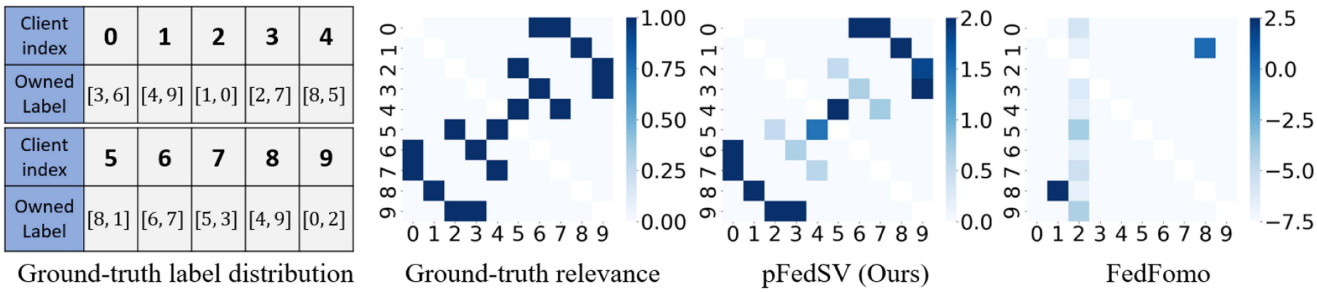


Fig. 7. The left chart shows the client label distribution obtained from the omniscience perspective. Right side is the visualization of clients' relevance matrix $\phi^i, i \in N$ on different algorithms with the pathological MNIST Non-IID setting after convergence. x-axis and y-axis is the client index.

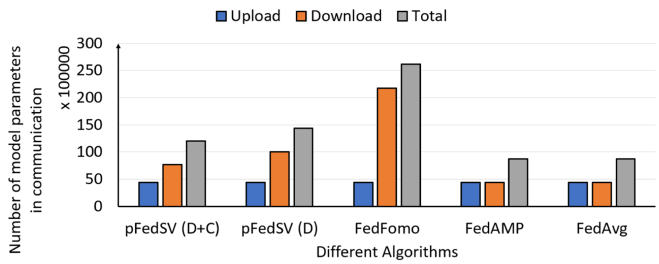


Fig. 8. LeNet-5: Communication overhead comparison on LeNet-5 with different algorithms. The y-axis indicates the number of model parameters in the communication.

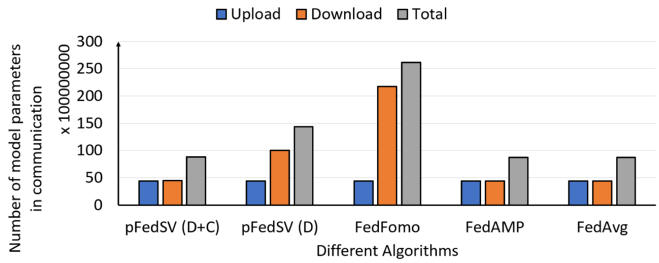


Fig. 9. ResNet: Communication overhead comparison on ResNet-V1-34-layer(Plain) with different algorithms. The y-axis indicates the number of model parameters in the communication.

convergence on different algorithms, where FedFomo uses the model similarity-based weights to update the relevance vector, while pFedSV use the computed SV from its local model coalition game. To illustrate the effectiveness of pFedSV algorithm, we also demonstrate the visualized ground-truth of client relevance according to the client label distribution table obtained from the omniscience perspective. Obviously, it is evident from ground-truth that symmetry is an important property of the client relevance matrix. Our pFedSV can perfectly identify all domain-relevant clients and assigns aggregation weights with multiwise collaboration in the coalition, while the FedFomo cannot guarantee a precise relevance identification.

C. Communication Overhead & Model Privacy

1) Communication Overhead Reduction: We have two different mechanisms in our paper to reduce the communication overhead: dynamic top-k download mechanism and shared common feature extractor. Therefore, to compare the communication overhead under different cases, we adopt the following baselines:

1) *pFedSV (D+C):* It means we adopt both the **D**ynamic top-k download mechanism and **C**ommon feature extractor in pFedSV to reduce the communication overhead.
2) *pFedSV (D):* It means we only adopt the **D**ynamic top-k download mechanism in the main content to reduce the communication overhead.

TABLE III
THE MTA COMPARISON OF PFEDSV WITH DP-BASED NOISE ADDITION AND SOME SELECTED BASELINES (OTHER OMITTED BASELINES CAN REFER TO TABLE II)

Methods	MNIST		FMNIST		CIFAR-10		CIFAR-100	
	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients	10 clients	100 clients
Seperate	74.05 ± 2.11	59.81 ± 5.73	60.18 ± 6.42	58.22 ± 6.73	40.53 ± 7.20	36.15 ± 6.88	35.43 ± 3.87	30.05 ± 5.49
FedAvg	43.57 ± 3.75	30.15 ± 4.82	40.58 ± 4.16	36.49 ± 5.07	33.81 ± 5.07	26.82 ± 6.43	26.17 ± 4.27	20.33 ± 5.27
FedFomo	72.54 ± 2.18	63.07 ± 2.54	64.75 ± 3.42	60.49 ± 3.72	53.83 ± 4.57	48.35 ± 5.29	45.91 ± 3.06	37.51 ± 3.09
FedAMP	70.15 ± 3.02	60.28 ± 3.11	62.28 ± 2.53	58.94 ± 3.14	51.57 ± 4.03	46.05 ± 4.48	43.67 ± 3.55	36.40 ± 3.76
pFedSV(Ours)	$\mathbf{78.17 \pm 1.59}$	$\mathbf{70.76 \pm 2.41}$	$\mathbf{71.47 \pm 1.86}$	$\mathbf{66.63 \pm 2.03}$	$\mathbf{61.18 \pm 1.67}$	$\mathbf{56.76 \pm 1.85}$	$\mathbf{50.46 \pm 2.47}$	$\mathbf{42.25 \pm 3.13}$
pFedSV+DP	$\mathbf{76.58 \pm 1.32}$	$\mathbf{68.43 \pm 1.86}$	$\mathbf{69.24 \pm 2.07}$	$\mathbf{65.31 \pm 1.95}$	$\mathbf{57.94 \pm 2.53}$	$\mathbf{53.28 \pm 1.66}$	$\mathbf{48.37 \pm 2.51}$	$\mathbf{40.79 \pm 2.84}$

The experiments are conducted with Dirichlet Non-IID data setting ($\alpha = 0.1$). 10 clients with 100% and 100 clients with 10% participation. We emphasize our pFedSV and the pFedSV+DP in bold.

TABLE IV
THE RESULTS OF PFEDSV WITH DP, WHICH ILLUSTRATES THAT WE CAN MAINTAIN THE PERSONALIZED ACCURACY WITH A REASONABLE PRIVACY BUDGET

Methods	δ	σ	CIFAR-10		CIFAR-100	
			ϵ	Accuracy	ϵ	Accuracy
FedAvg	1×10^{-5}	0	∞	19.68 ± 1.76	∞	5.21 ± 0.41
FedAvg	1×10^{-5}	1	11.28 ± 0.32	17.54 ± 1.37	8.47 ± 0.67	5.03 ± 0.24
FedAvg	1×10^{-5}	2	3.64 ± 0.13	15.97 ± 1.53	2.56 ± 0.19	4.37 ± 0.19
pFedSV	1×10^{-5}	0	∞	84.73 ± 1.67	∞	31.07 ± 1.22
pFedSV	1×10^{-5}	1	$\mathbf{5.97 \pm 0.11}$	$\mathbf{82.16 \pm 1.55}$	$\mathbf{8.42 \pm 0.71}$	$\mathbf{30.59 \pm 1.06}$
pFedSV	1×10^{-5}	2	1.82 ± 0.05	78.29 ± 1.63	1.80 ± 0.16	23.44 ± 0.89

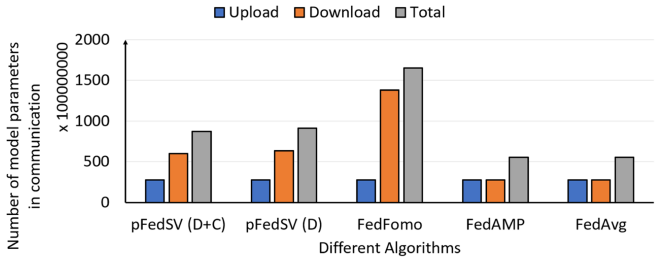


Fig. 10. VGG-19: Communication overhead comparison on VGG-19 with different algorithms. The y-axis indicates the number of model parameters in the communication.

3) *FedFomo:* It downloads the whole model of other clients and performs personalization on the local side of each client [5].
4) *FedAMP:* it performs the personalization on the server side and directly distributes the personalized model to each client, whose communication overhead is equal to FedAvg [6].
5) *FedAvg:* traditional FL algorithm that downloads one global model to each client [3].

All algorithms are implemented with the following setups: total 20 communication rounds, 10 clients with 100% participation in each round, pathological Non-IID data distribution. We use the number of model parameters that are required in upload and download as the measurement metric for communication overhead.

In Fig. 8, we show the communication overhead comparison of different baselines on the LeNet-5 Model. Besides, to further illustrate the effectiveness of our Top-k dynamic download mechanism and shared common feature extractor in communication overhead reduction, we also compute the communication overhead comparison on other different models, including ResNet-V1-34-layer(Plain) in Fig. 9 and VGG-19 in Fig. 10.

You can find that the communication overhead of pFedSV at ResNet case is almost the same as traditional FedAvg. The reason is that, as a powerful pre-trained model, most model parameters in ResNet are the convolutional layer-based feature extractor, and the classifier-related parameters only account for 2.3% of the overall model parameter number. Thus using a shared common feature extractor can significantly save extensive communication overhead. In contrast, for traditional CNN model such as LeNet-5, the classifier-related parameters can account for 49.57% of the overall model parameter number. Therefore, with the help of shared feature extractor, the additional communication can be significantly reduced in ResNet case. Moreover, the results on VGG-19 are for your additional reference, where the classifier-related parameters can account for 89.74% of the overall model parameter number in VGG-19.

As expected, the communication overhead of FedFomo is much higher than other algorithms. Our dynamic download mechanism can efficiently reduce it by rapidly identifying the domain-relevant clients and adjusting the model download number, which is illustrated in the main content. Besides, the introduced common feature extractor can further reduce the communication overhead in the download part. Finally, FedAMP has the same communication overhead as FedAvg. Although the communication overhead of our pFedSV (D+C) is not the lowest compared to FedAMP, we can achieve higher personalized performance for each client, which is an acceptable trade-off.

2) Model Privacy Protection: We consider a task with the pathological data Non-IID setting on CIFAR-10 and CIFAR-100 dataset, with 10 clients and 100% participation at each round. We compare pFedSV with the FedAvg under different levels of Gaussian noise σ, and all other parameters are fixed. The results in Table IV indicates that a higher σ leads to improved privacy (lower ϵ) at the cost of decreased performance. The bold value in the table indicates that pFedSV can still achieve promising accuracy when providing enough privacy protection. The experiment results in Table IV have shown that adding aggressive noise will cause accuracy reduction (from 84.73% to 78.29%). However, adopting an appropriate noise ($\delta = 1$) can also protect the model privacy with only a minor impact on accuracy (from 84.73% to 82.16%). We also conduct additional experiments to show that, under an appropriate noise ($\delta = 1$), the performance of our pFedSV+DP can still outperform other personalized baselines, where the results can be found in Table III.

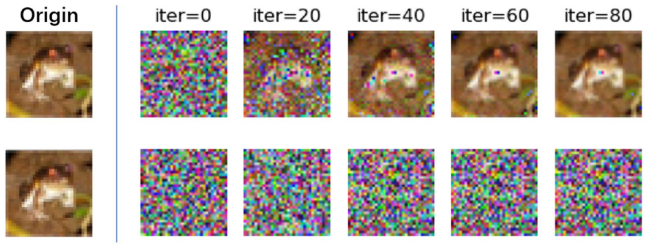


Fig. 11. The model inversion attack results on the original model and the model with DP noise addition ($\delta = 1$) on the CIFAR-10 dataset. *The top column* is the original model, where the attacker can recover the raw training data with the shared model parameters. *The bottom column* is the model with DP noise addition, where the attack failed.

Therefore, the DP-based methods are still effective in solving privacy issues, which are widely validated by many works.

Besides, the key privacy issue concern of our pFedSV algorithm comes from the fact that the local model of each client will be downloaded to other clients, since the recent research [42], [43], [44] on model inversion attacks shows that malicious attackers can recover the raw training data from the model through the gradients only. Therefore, to further demonstrate the effectiveness of our DP-based noise addition on privacy protection, we conduct an extra experiment with model inversion attack on both the original local model and the model with DP-based noise addition, where the results are illustrated in Fig. 11. We can see that the attacker cannot recover the raw training data after we add the DP-based noise into the original model.

VII. Conclusion

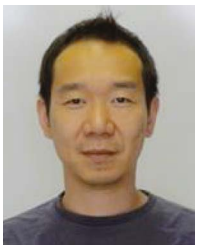
In this paper, we focus on the model personalization of clients with heterogeneous domains in an agnostic federated learning system. we propose pFedSV, a novel personalized FL algorithm that incorporates the Shapley value from coalition game theory to assess intricate, multi-faceted influences by quantifying the individual contributions of each client. We provide a complex analysis by formulating the model aggregation process as a coalition game, which not only helps form the personalized domain-relevant coalition but also serves as personalized aggregation weights for each client. Extensive experiments are conducted to demonstrate the effectiveness of pFedSV and the results empirically illustrate its superiority through the significant improvement on personalized accuracy. Furthermore, regarding the communication overhead and model privacy issues raised by the local model download mechanism in pFedSV, we introduce the shared common feature extractor and the DP-based noise addition, respectively.

References

[1] P. Kairouz et al., "Advances and open problems in federated learning," 2019, *arXiv: 1912.04977*.
[2] T. Guo, S. Guo, J. Wang, X. Tang, and W. Xu, "PromptFL: Let federated participants cooperatively learn prompts instead of models-federated learning in age of foundation model," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5179–5194, May 2024.
[3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Conf. Artif. Intell. Statist.*, PMLR, 2017, pp. 1273–1282.
[4] A. Shamsian, A. Navon, E. Fetaya, and G. Chechik, "Personalized federated learning using hypernetworks," 2021, *arXiv:2103.04628*.
[5] M. Zhang, K. Sapra, S. Fidler, S. Yeung, and J. M. Alvarez, "Personalized federated learning with first order model optimization," in *Proc. Int. Conf. Learn. Representations*, 2020.
[6] Y. Huang et al., "Personalized cross-silo federated learning on non-IID data," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 7865–7873.
[7] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," 2021, *arXiv:2102.07078*.
[8] K. Donahue and J. Kleinberg, "Model-sharing games: Analyzing federated learning under voluntary participation," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 5303–5311.
[9] A. Z. Tan, H. Yu, L. Cui, and Q. Yang, "Towards personalized federated learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 9587–9603, Dec. 2023.
[10] Y. Tong et al., "Federated computing: Query, learning, and beyond," *IEEE Data Eng. Bull.*, vol. 46, pp. 9–26, 2023.
[11] T. Guo, S. Guo, and J. Wang, "pFedPrompt: Learning personalized prompt for vision-language models in federated learning," in *Proc. ACM Web Conf.*, 2023, pp. 1364–1374.
[12] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, "Three approaches for personalization with applications to federated learning," 2020, *arXiv: 2002.10619*.
[13] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, and D. Ramage, "Federated evaluation of on-device personalization," 2019, *arXiv: 1910.10252*.
[14] C. T. Dinh, N. Tran, and T. D. Nguyen, "Personalized federated learning with moreau envelopes," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 21394–21405, 2020.
[15] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proc. Mach. Learn. Syst.*, vol. 2, pp. 429–450, 2020.
[16] Y. Jiang, J. Konečnỳ, K. Rush, and S. Kannan, "Improving federated learning personalization via model agnostic meta learning," 2019, *arXiv: 1909.12488*.
[17] O. Marfoq, G. Neglia, A. Bellet, L. Kameni, and R. Vidal, "Federated multi-task learning under a mixture of distributions," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 15 434–15 447, 2021.
[18] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 19 586–19 597, 2020.
[19] D. Fudenberg and J. Tirole, *Game Theory*. Cambridge, MA, USA: MIT Press, 1991.
[20] G. Wang, C. X. Dang, and Z. Zhou, "Measure contribution of participants in federated learning," in *Proc. IEEE Int. Conf. Big Data*, 2019, pp. 2597–2604.
[21] G. Wang, "Interpret federated learning with shapley values," 2019, *arXiv: 1905.04519*.
[22] T. Song, Y. Tong, and S. Wei, "Profit allocation for federated learning," in *Proc. IEEE Int. Conf. Big Data*, 2019, pp. 2577–2586.
[23] H. Yu et al., "A fairness-aware incentive scheme for federated learning," in *Proc. AAAI/ACM Conf. AI Ethics Soc.*, 2020, pp. 393–399.
[24] L. Nagalapatti and R. Narayanam, "Game of gradients: Mitigating irrelevant clients in federated learning," 2021, *arXiv:2110.12257*.
[25] Q. Sun et al., "ShapleyFL: Robust federated learning based on shapley value," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 2096–2108.
[26] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-IID data," 2018, *arXiv: 1806.00582*.
[27] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of FedAVG on non-IID data," in *Proc. Int. Conf. Learn. Representations*, 2019.
[28] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2020, pp. 5132–5143.
[29] I. Mann and L. S. Shapley, "Values of large games. 6: Evaluating the electoral college exactly," Rand Corp, Santa Monica CA, USA, Tech. Rep. RM-3158-PR, 1962.
[30] J. Castro, D. Gómez, and J. Tejada, "Polynomial calculation of the shapley value based on sampling," *Comput. Operations Res.*, vol. 36, no. 5, pp. 1726–1730, 2009.

[31] S. Maleki, L. Tran-Thanh, G. Hines, T. Rahwan, and A. Rogers, "Bounding the estimation error of sampling-based shapley value approximation," 2013, *arXiv:1306.4265*.
[32] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Mach. Learn.*, vol. 79, no. 1, pp. 151–175, 2010.
[33] M. Luo, F. Chen, D. Hu, Y. Zhang, J. Liang, and J. Feng, "No fear of heterogeneity: Classifier calibration for federated learning with non-IID data," 2021, *arXiv:2106.05001*.
[34] Y. LeCun, "The MNIST database of handwritten digits," 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/
[35] M. Abadi et al., "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 308–318.
[36] H. B. McMahan, E. Moore, D. Ramage, and B. A. Y. Arcas, "Federated learning of deep networks using model averaging," 2016, *arXiv:1602.05629*.
[37] P. P. Liang et al., "Think locally, act globally: Federated learning with local and global representations," 2020, *arXiv: 2001.01523*.
[38] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv: 1708.07747*.
[39] A. Krizhevsky et al., Learning multiple layers of features from tiny images, 2009.
[40] Z. Zhu, J. Hong, and J. Zhou, "Data-free knowledge distillation for heterogeneous federated learning," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2021, pp. 12 878–12 889.
[41] X. Zhang, M. Hong, S. Dhople, W. Yin, and Y. Liu, "FedPD: A federated learning framework with optimal rates and adaptivity to non-IID data," 2020, *arXiv: 2005.11418*.
[42] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song, "The secret revealer: Generative model-inversion attacks against deep neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 253–261.
[43] J. Wang, S. Guo, X. Xie, and H. Qi, "Protect privacy from gradient leakage attack in federated learning," in *Proc. IEEE Conf. Comput. Commun.*, 2022, pp. 580–589.
[44] R. Zhang, S. Guo, J. Wang, X. Xie, and D. Tao, "A survey on gradient inversion: Attacks, defenses and future directions," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, 2023, pp. 5678–685.

Leijie Wu (Student Member, IEEE) received the BEng degree of science in automation from the School of Xuteli, Beijing Institute of Technology, Beijing, China, in 2019. He is currently working toward the PhD degree with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China. His current research interest includes federated learning, mobile edge computing, deep reinforcement learning, and incentive mechanism design.



Song Guo (Fellow, IEEE) is a full professor with the Department of Computing, The Hong Kong Polytechnic University. He also holds a Changjiang Chair Professorship awarded by the Ministry of Education of China. He is a fellow of the Canadian Academy of Engineering. His research interests are mainly in edge AI, machine learning, mobile computing, and distributed systems. He published many papers in top venues with wide impact in these areas and was recognized as a Highly Cited Researcher (Clarivate Web of Science). He is the recipient of more than a dozen Best Paper Awards from IEEE/ACM conferences, journals, and technical committees. He is the editor-in-chief of IEEE Open Journal of the Computer Society and the Chair of IEEE Communications Society (ComSoc) Space and Satellite Communications Technical Committee. He was an IEEE ComSoc distinguished lecturer and a member of IEEE ComSoc Board of Governors. He has served for IEEE Computer Society on Fellow Evaluation Committee, and been named on editorial board of a number of prestigious international journals like *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Cloud Computing*, *IEEE Transactions on Emerging Topics in Computing*, etc. He has also served as chairs of organizing and technical committees of many international conferences.

Yaohong Ding (Student Member, IEEE) is currently working toward the BSc CS degree with the Department of Computing, The Hong Kong Polytechnic University. His research interests include federated learning and edge computing.

Junxiao Wang received the PhD degree from the Dalian University of Technology, China, in 2020. He is currently an associate professor with Guangzhou University. Before that, he was a researcher with King Abdullah University of Science and Technology, The Hong Kong Polytechnic University and Queen Mary University of London. He is broadly interested in artificial intelligence and system with a special focus on distributed machine learning, AI security, privacy and interpretability. His research papers were published in many prestigious journals and conferences including *IEEE Transactions on Mobile Computing*, INFOCOM, WWW, KDD, NeurIPS, ICLR, IJCAI and CVPR, etc.

Wenchao Xu (Member, IEEE) received the BE and ME degrees from Zhejiang University, Hangzhou, China, in 2008 and 2011, respectively and the PhD degree from the University of Waterloo, Canada, in 2018. He is a research assistant professor with The Hong Kong Polytechnic University. In 2011, he joined Alcatel Lucent Shanghai Bell Company Ltd., where he was a software engineer for telecom virtualization. He has also been an Assistant Professor with the School of Computing and Information Sciences in Caritas Institute of Higher Education, Hong Kong. His research interests includes wireless communication, Internet of Things, distributed computing and AI enabled networking.

Yufeng Zhan received the PhD degree from the Beijing Institute of Technology (BIT), Beijing, China, in 2018. He is currently an associate professor with the School of Automation with BIT. Prior to join BIT, he was a postdoctoral fellow with the Department of Computing with The Hong Kong Polytechnic University. His research interests include networking systems, game theory, and machine learning.

Anne-Marie Kermarrec is professor with EPFL since 2020. Before that she was the CEO of the Mediego startup that she founded in 2015. Mediego provides content personalization services for online publishers. She was a research director with Inria, France from 2004 to 2015. She got a PhD thesis from the University of Rennes (France), and has been with Vrije Universiteit, NL and Microsoft Research Cambridge, U.K. She received an ERC grant in 2008 and an ERC Proof of Concept in 2013. She received the Montpetit Award in 2011 and the Innovation Award in 2017 from the French Academy of Science. She has been elected to the European Academy in 2013 and named ACM fellow in 2016. Her research interests are in large-scale distributed systems, epidemic algorithms, peer to peer networks and distributed learning systems.