PPFL: A Parameter Behavior-Driven Plug-in Personalization Engine for Federated Learning

Qianyue Cao[1], Zongwei Zhu[1,2*], Zirui Lian[1], Rui Zhang[1], Boyu Li[1], Yi Xiong[1], Xuehai Zhou[1]

[1]University of Science and Technology of China
[2]Suzhou Institute of Advanced Research, University of Science and Technology of China
{cqy_1999, lzrustc, ruizhang007, llbbyy, xiongyi}@mail.ustc.edu.cn, {zzw1988, xhzhou}@ustc.edu.cn

Abstract

Personalized Federated Learning (PFL) customizes models for each client to mitigate challenges from non-IID data, wherein a dominant strategy is model decoupling that partitions models into shared and personalized parts based on architectural priors (e.g., backbone vs. head). However, we reveal a critical flaw in this strategy: it induces "intrinsic drift," a performance degradation often more severe than the well-known client drift, which limits final accuracy. We trace this drift to a steep cliff of high loss emerging from the naive stitching of shared and personalized parts. To address this, we shift from architectural partitioning to a parameter behavior-driven paradigm. We introduce PPFL, an approach that employs a novel soft-fusion strategy guided by parameter-wise behavioral perception. PPFL dynamically infers each parameter's functional role—whether it behaves more like a 'personalist' or a 'generalist' in the current context—by synthesizing its multifaceted behavior observed during local training. Extensive experiments on image, text, and multimodal classification benchmarks show that PPFL outperforms eight state-of-the-art baselines by up to 5.3%. Moreover, it can function as a plug-in module, boosting the accuracy of vanilla FedAvg with a 16.82% absolute gain.

Introduction

Federated Learning (FL) (McMahan et al. 2017) has emerged as a privacy-preserving paradigm that enables collaborative model training across decentralized clients without exposing their local data. However, data across clients is commonly non-identically distributed (non-IID) (Hsieh et al. 2020) due to variations in geographic regions, devices, and user behaviors. Traditional FL (TFL) typically aims to learn a single global model that generalizes across all clients (Li et al. 2023). However, a single global model is ill-suited for highly heterogeneous data settings, where it struggles to generalize across diverse client distributions and leads to suboptimal or unfair outcomes (Zhang et al. 2021a; Wang et al. 2021). To address these limitations, Personalized Federated Learning (PFL) (Tan et al. 2023) has gained traction as a solution, where each client is allowed to obtain a customized local model tailored to its unique data distribution after participating in the federated training process.

*Corresponding author.

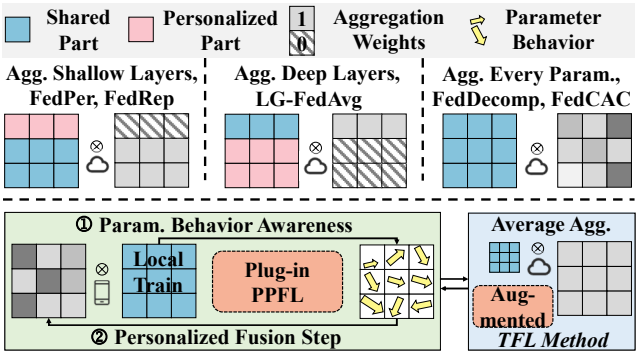


Figure 1: **(Above)** Classification of PFL approaches. **(Below)** Our PPFL method via parameter behavior fusion.

Status quo and limitations. A prominent paradigm in PFL is model decoupling, which partitions a model's parameters into shared and personalized parts (Zhang et al. 2023d). This approach bases the partition on the model's architecture—for example, by sharing a feature extractor to learn general knowledge while personalizing a classification head for client-specific adaptation (Arivazhagan et al. 2019; Collins et al. 2021; Chen and Chao 2022). However, the assumption of this approach—that a parameter's functional role is dictated by its architectural position—is a flawed oversimplification. A growing body of evidence reveals a significant disconnect between architectural structure and parameter function. Parameters with distinct learning dynamics (Raghu et al. 2017) and functional roles (Bengio 2012)—capturing either general patterns or client-specific nuances—are often entangled within the same layer (Maini et al. 2023). This functional entanglement explains why even advanced methods that identify and selectively average entire 'aggregation-sensitive' layers yield only limited personalization gains (Adilova et al. 2024). Such a rigid, architecture-based split is particularly brittle in realistic PFL scenarios involving client resource heterogeneity and dynamic participation, where parameter behavior becomes even more complex and unpredictable.

Motivation. To understand the consequences of functional entanglement, we study impact during the local aggregation step in PFL. We find that simply "stitching" the

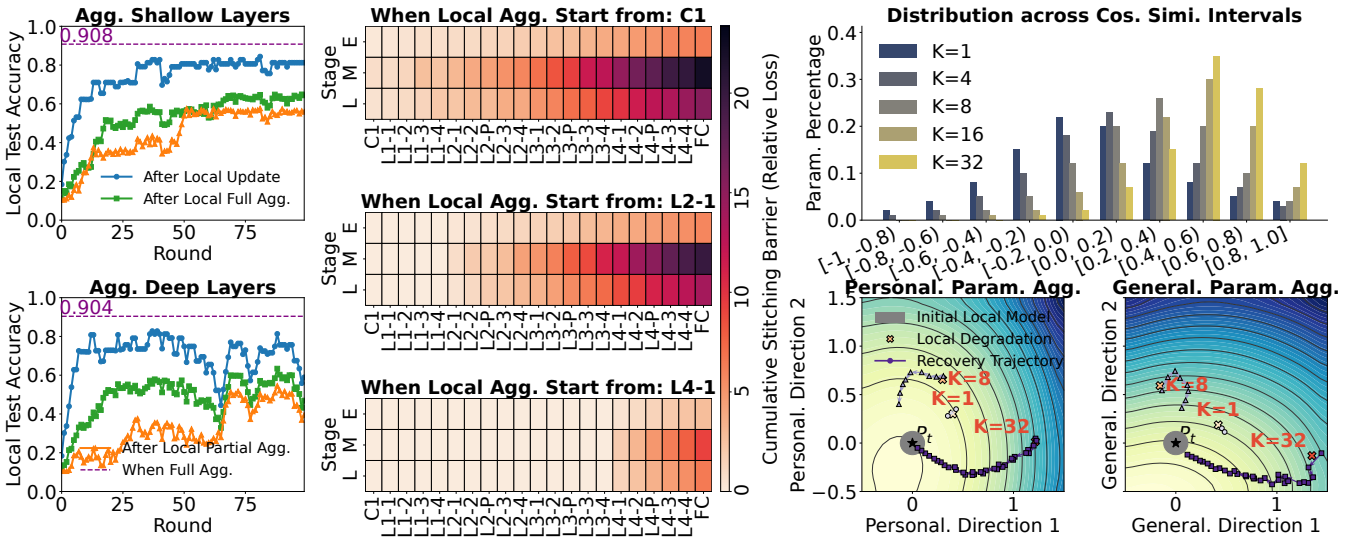


Figure 2: Pilot experiments using SimpleCNN/ResNet-18 on a practical non-IID CIFAR-10/100 dataset support our claims. **(Left)** The effect of partial aggregation on shallow and deep layers. **(Mid)** The cumulative barrier in model decoupling. **(Right)** Parameter behavior and optimization trajectories.

shared part to the personalized part creates a significant loss barrier at the partition interface. This barrier persists and remains strong regardless of where the partition occurs—whether early in the feature extractor or late in the classification head. This suggests that the core issue lies not in the partition location but in the rigid binary partitioning imposed by a static architectural prior. To explore whether the loss barrier is an inevitable static result of model decoupling or a controllable dynamic conflict, we introduce a method to control the state of the local model before it conflicts with the global aggregation. Extending local iterations K is the most direct way to mature the representations learned by the local model. By increasing the number of local iterations, we significantly weakened the loss barrier, demonstrating that the barrier is not a fixed structural flaw but a dynamic conflict. The conflict arises when the global model overwrites local parameters that have not had enough time to stabilize in their functional role. This reveals the true source of the conflict: it is not the layer, but the individual parameters that have found their purpose.

This inspires a shift from the static question of whether a parameter should aggregate to the dynamic question of how much it should aggregate based on its behavior.

Our solutions. We propose PPFL, a novel PFL framework that replaces the paradigm of model decoupling with behaviorally-driven soft-fusion. PPFL exploits this enhanced flexibility through two coordinated client steps: **1) Parameter Behavior Awareness Step:** PPFL operates during local training to generate personalized scores for each parameter by integrating sensitivity, update direction consistency, and activation metrics. The resulting score map serves as a fingerprint of the client's personalization needs at the current train stage. **2) Personalized Fusion Step:** The score map directly governs the client's update process via an adaptive fusion controller. Upon receiving the global model, this "actuator" performs parameter-wise fusion guided by the personalization scores. This mechanism uses the score as an intelligent weighting factor, ensuring that each client's personalized core is preserved while seamlessly integrating transferable global knowledge. More importantly, PPFL serves as **a generic and pluggable personalization engine.** Since the entire process is executed locally on the client, it remains fully agnostic to the server-side aggregation logic. Whether the server employs FedAvg (McMahan et al. 2017), FedProx (Li et al. 2020a), or any other global update rule, PPFL can seamlessly encapsulate the client update step.

Contributions of this work are summarized as follows:

- We identify the phenomenon of intrinsic drift in model decoupling methods in PFL and show that this challenge manifests as a loss barrier at the partition interface.
- We introduce a personalization paradigm that shifts from static, layer-wise to dynamic, parameter-wise decisions, mitigating client and intrinsic drift.
- Experimental results on various benchmarks show that PPFL outperforms eight SOTA methods by up to 5.3% in accuracy and improves the absolute accuracy of vanilla FedAvg by 16.82% as a personalization plug-in.

Related Work

Decoupling-Based Personalization A dominant paradigm in PFL is model decoupling, which splits model parameters into a globally shared part and a locally personalized one. Canonical methods such as FedPer (Arivazhagan et al. 2019) and FedRep (Collins et al. 2021) adopt a shared feature extractor for global training and personalize the classification head. This principle underlies many variants: FedBABU (Oh, Kim, and Yun 2021) fine-tunes a local head after global body training, while FedGH (Yi et al. 2023) generalizes to heterogeneous client via a global header.

To overcome the limitations of rigid head-body separation, later work introduces more flexible decoupling. FedRoD (Chen and Chao 2022) and FedAS (Yang, Huang, and Ye 2024) add auxiliary objectives or classifiers to balance generalization and personalization. FedCP (Zhang et al. 2023c) and GPFL (Zhang et al. 2023a) further disentangle generic and personalized features—sometimes within a single block—via conditional policies or specialized branches.

Interpolation-Based Personalization This line of work personalizes models by interpolating between a global model and a local model, aiming to balance generalization and specialization. A seminal approach, APFL (Deng, Kamani, and Mahdavi 2020), learns a scalar coefficient to linearly combine the two models into a personalized one.

Building on this concept, SuPerFed (Hahn, Jeong, and Lee 2022) and Floco (Grinwald, Wiesner, and Nakajima 2024) adopt a geometric view, seeking personalized models within a connected low-loss subspace beyond the global-local line. Alternatively, FedFomo (Zhang et al. 2021b) and APPLE (Luo and Wu 2022) apply meta-learning, implicitly achieving personalization through fast local adaptation via gradient-based updates from the global model.

Parameter-wise Aggregation-Based Personalization Several works move from coarse layer-level personalization to fine-grained, parameter-wise approaches. These methods assess each parameter's contribution to the global model, determining how much global knowledge it should absorb. FedALA (Zhang et al. 2023b) learns continuous, per-parameter aggregation weights via gradient descent. To balance computational overhead, FedALA is to learn aggregation weights only for the model's classification head. FedCAC (Wu et al. 2023) classifies parameters as "consensus" or "controversial," using a client similarity metric based on data distribution, under the assumption that similar distributions yield more consensus parameters. However, this strategy may conflict with strict privacy principles.

Background and Motivation

Personalized Federated Learning Consider a federated system with N clients, indexed by $n \in 1, 2, \ldots, N$. Each client n holds a private dataset $\mathcal{D}_n$ with $|\mathcal{D}_n|$ samples drawn from a local distribution $\mathcal{P}_n(x, y)$, where (x, y) denotes a feature-label pair. A key assumption in PFL is statistical heterogeneity across clients, i.e., $\mathcal{P}_n \neq \mathcal{P}_m$ for $n \neq m$.

Unlike standard federated learning, which learns a global model w, PFL aims to learn personalized models ${w_n}_{n=1}^N$, with each $w_n \in \mathbb{R}^d$ tailored to client n's data. The performance of w_n is measured by the local objective:

$$\mathcal{L}_n(w_n) := \mathbb{E}_{(x,y)\sim\mathcal{P}_n}[\ell(w_n; x, y)], \tag{1}$$

where $\ell(\cdot)$ is the local loss function, approximated in practice by empirical loss over $\mathcal{D}_n$. The global objective is to minimize the average local loss:

$$\min_{w_1,w_2,\ldots,w_N} \left\{ \mathcal{F}(w_1, \ldots, w_N) := \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}_n(w_n) \right\}, \tag{2}$$

Limitations of Model Decoupling This paradigm often assumes that a parameter's function is determined by its architectural position, simplifying the model into a shared feature extractor (backbone) and a personalized classifier (head). Recent studies challenge this oversimplification. For instance, research on linear mode connections (Adilova et al. 2024) shows that different layers within the same backbone exhibit distinct connectivity characteristics, suggesting that a global view of the feature extractor is insufficient. Moreover, the notion that personalized knowledge is confined to the classifier head has been questioned, with evidence (Maini et al. 2023) indicating that such knowledge is distributed throughout the model. In the context of model decoupling for PFL, this flawed assumption leads to a detrimental consequence: performance degradation, often worse than the drift observed in full-model average aggregation. To show this, we conducted a pilot experiment (see Appendix for details). As shown in Fig. 2, after local aggregation, inference accuracy drops significantly for both `conv-1,2` and `fc-1,2` layers. Compared to full-model aggregation, two key points stand out: first, accuracy drops more noticeably, with an average decline of 10.43%; second, after 100 training epochs, final accuracy drops by 2.75-4.37%.

Intrinsic Drift as the Root Cause To understand the cause of this performance drop, we apply the Layer-wise Linear Mode Connection (LLMC) theory (Adilova et al. 2024). From this perspective, we treat the model before and after local aggregation as two distinct points in parameter space, with personalized parts identical. Interpolating between them reveals a high-loss barrier during the decoupling process. As shown in Fig. 2, starting interpolation at `conv-1` leads to a 25.1% higher final `fc` loss compared to

Algorithm 1: Pluggable PFL: PPFL

Input: Communication rounds T, local steps K, global model $w_{\mathcal{G}}$, client models $\{w_i\}_{i=1}^N$

Server executes:
Initialize $w_{\mathcal{G}}^{(0)}$; Clients initialize $w_i^{(0)} \leftarrow w_{\mathcal{G}}^{(0)}$
for each round $t = 0, 1, \ldots, T-1$ **do**
  Select a subset of clients S_t
  Broadcast $w_{\mathcal{G}}^{(t)}$ to clients in S_t
  for each client $i \in S_t$ **in parallel do**
    $\Delta_i, w_i^{(t+1)} \leftarrow \text{ClientUpdate}(i, w_{\mathcal{G}}^{(t)}, w_i^{(t)})$
    $\Delta_{\text{agg}} \leftarrow \Delta_{\text{agg}} + \Delta_i$
  end for
  $w_{\mathcal{G}}^{(t+1)} \leftarrow w_{\mathcal{G}}^{(t)} + \frac{1}{|S_t|}\Delta_{\text{agg}}$
end for
return $\{w_i^{(T)}\}_{i=1}^N$

function ClientUpdate($i, w_{\mathcal{G}}, w_i^{\text{prev}}$)
$w_i^{\text{local}} \leftarrow \text{LocalTrain}(w_i^{\text{prev}}, \mathcal{D}_i, K)$
*// — **Param. Behavior Awareness Step** —*
$E_i \leftarrow \text{GatherAdaptiveEvidence}(w_i^{\text{local}}, w_i^{\text{prev}}, K)$
$\hat{\rho}_i \leftarrow \text{ComputeBehavioralScore}(E_i)$ {Eq. (3)-(9)}
*// — **Personalized Fusion Step** —*
$\Phi_j^{\text{next}} \leftarrow (1 - \hat{\rho}_j) \odot \Phi_{\mathcal{G}} + \hat{\rho}_j \odot \Phi_j^{\text{local}}$ {Eq. (11)}
$\Delta_i \leftarrow w_i^{\text{local}} - w_{\mathcal{G}}$
return $\Delta_i, w_i^{\text{next}}$
end function

starting at `layer 4-1`. Early in training, aggregation from shallow layers causes the largest loss barrier, but as training progresses, these barriers converge. We propose two hypotheses: first, the success of methods treating `fc` layers as a personalized head may succeed due to aggregation of most parameters, rather than precise generalization in the backbone. Second, as training progresses, different parts of the model capture both foundational and advanced knowledge related to local data, and partial aggregation can disrupt this representation. Thus, model decoupling causes a intrinsic drift in the model, leading to greater drift.

The Principle of Behavior-Driven Aggregation We observe that increasing local update iterations K (reducing batch size while maintaining the same total training data per epoch) effectively mitigates the performance drop after partial aggregation in model decoupling (details in Appendix). To investigate the cause, we analyze parameter behavior during local updates. First, we examine the cosine similarity distribution of updates. As shown in Fig. 2(c), increasing K transforms updates from a high-variance, zero-centered random walk (K=1, mean=0.014) to a stable, directional trajectory (K=32, mean=0.526). We then partition parameters into high-consistency "personalization" and low-consistency "generalization" groups and visualize the loss landscapes along both directions. The landscape shows that larger K ensures stable descent along the personalization direction, while the generalization direction exhibits exploratory behavior across multiple loss basins, highlighting the coordination of volatile updates. This trajectory likely reflects the model's recalibration of internal parameter synergy, driven by partial aggregation.

Inspired by the above analysis, we shift from exploring the static, binary question of whether a parameter should aggregate, to investigating the dynamic, nuanced question of to what extent it should be aggregated based on its behavior.

Method

Design Principles: Exact per-parameter Bayesian inference is computationally prohibitive in FL. We develop a lightweight yet efficient heuristic framework to score each parameter's propensity for personalization (see Appendix for details). First, we investigate a broad set of candidate metrics spanning dynamic update trajectories and static properties, screening them for effectiveness and efficiency. Second, we employ correlation analysis to distill a compact, complementary set of indicators, thereby eliminating redundancy and minimizing overhead. Finally, we evaluate various fusion mechanisms and select the optimal one.

Parameter Behavior Awareness The goal of this step is to infer the posterior probability that a parameter belongs to the "personalization set," based on behavioral evidence observed during local training. Specifically, we model a parameter Φ_j as a latent random variable Z_j, which can take two states: $\mathbb{G}$ (Generalist) or $\mathbb{P}$ (Personalist). The objective is to compute $\rho_j = P(Z_j = \mathbb{P}|E_j)$, where E_j is the behavioral evidence vector for Φ_j.

We propose a unified evaluation method to measure parameter's personalization tendency during the training phase. This tendency can be observed from two perspectives: dynamic trajectories and static properties. First, we analyze the dynamic trajectory, which reflects how they evolve toward the current model state based on client-specific data. We use dynamic magnitude v_j and fidelity f_j to capture this trajectory. For magnitude v_j

$$\begin{aligned} v_j &\triangleq \left\| \sum_{k=1}^{K} \left(\Phi_j^{(t,k)} - \Phi_j^{(t,k-1)} \right) \right\|_2 \\ &= \left\| \left(\Phi_j^{(t,1)} - \Phi_j^{(t,0)} \right) + \left(\Phi_j^{(t,2)} - \Phi_j^{(t,1)} \right) \right. \\ &\quad \left. + \cdots + \left(\Phi_j^{(t,K)} - \Phi_j^{(t,K-1)} \right) \right\|_2 \\ &= \left\| \Phi_j^{(t,K)} - \Phi_j^{(t,0)} \right\|_2, \end{aligned} \tag{3}$$

Considering the magnitude of updates is insufficient. Truly personalized parameters should steadily converge to a local optimum, rather than fluctuating erratically. The fidelity f_j is the normalized cosine similarity between the current update

$$f_j \triangleq \frac{1 + \cos_{\text{sim}}(\Delta\Phi_j^{(t)}, \mu_j^{(t-1)})}{2}, \tag{4}$$

where $\mu_j^{(t-1)}$ as the average of all past update vectors:

$$\begin{aligned} \mu_j^{(t-1)} &= \frac{1}{t-1} \sum_{i=1}^{t-1} \Delta\Phi_j^{(i)} \\ &= \frac{1}{t-1} \left(\Delta\Phi_j^{(1)} + \Delta\Phi_j^{(2)} + \cdots + \Delta\Phi_j^{(t-1)} \right) \\ &= \frac{S_j^{(t-1)}}{t-1}, \quad \text{where} \quad S_j^{(t-1)} = \sum_{i=1}^{t-1} \Delta\Phi_j^{(i)}, \end{aligned} \tag{5}$$

To avoid storing all past updates, we compute the historical trend recursively. At round t, we update the running sum $S_j^{(t)}$ as the sum of the previous sum and the current update $\Delta\Phi_j^{(t)}$. This recursive approach ensures efficient computation with minimal memory and overhead.

Next, we assess the parameter's static attributes, which provide a complementary snapshot of its functional role and importance for the local data. This offers evidence independent of the optimization path, focusing on the parameter's inherent properties at a single point in time. The parameter's potential for influence is quantified by saliency m_j, while its pattern of engagement is characterized by specificity s_j.

The saliency m_j is measured by its impact on the local loss function. Following FedCAC (Wu et al. 2023), we approximate this impact using a first-order Taylor expansion.

$$m_j \triangleq \|\nabla_{\Phi_j} \mathcal{L}_i(\mathbf{w})\|_2 \cdot \|\Phi_j\|_2, \tag{6}$$

We measure specificity s_j through the sparsity of its activation map $\mathbf{A}_j$. This metric reflects the parameter's activation policy: $\mathbb{G}$ interacts with a broad range of inputs, while $\mathbb{P}$ remains inactive until relevant features appear.

$$s_j \triangleq 1 - \mathbb{E}_{\mathbf{x} \in \mathcal{D}_i} \left[\frac{\|\mathbf{A}_j(\mathbf{x})\|_0}{|\mathbf{A}_j|} \right], \tag{7}$$

The parameter's propensity for personalization is high only if it shows strong evidence across all behavioral dimensions. We thus encapsulate these raw measurements into a behavioral evidence vector $E_j \triangleq \mathcal{I}(v_j, f_j, m_j, s_j)$. To ensure equal contribution from all metrics, we normalize unbounded metrics (v_j and m_j) using a min-max scaling:

$$\mathcal{N}(x_j) = \frac{x_j - \min_k(x_k)}{\max_k(x_k) - \min_k(x_k) + \epsilon}, \quad x \in \{v, m\} \quad (8)$$

where ϵ is a small constant, ensuring numerical stability. We obtain the final evidence vector E_j:

$$E_j = \mathcal{I}_j = \underbrace{\mathcal{N}(v_j) \cdot f_j}_{\text{Dynamic Score}} \cdot \underbrace{\mathcal{N}(m_j) \cdot s_j}_{\text{Static Score}}, \quad (9)$$

With the adaptively defined evidence vector E_j, we can formally express the posterior probability of a parameter being a $\mathbb{P}$ using Bayes' theorem:

$$\rho_j = P(Z_j = \mathbb{P} \mid E_j) = \frac{P(E_j \mid Z_j = \mathbb{P})P(Z_j = \mathbb{P})}{P(E_j)}, \quad (10)$$

The prior $P(Z_j = \mathbb{P})$ represents our initial belief about a parameter's identity. To avoid introducing external interference and to allow the behavioral evidence to be the primary driver of the inference, we assume a non-informative (uniform) prior. Consequently, the posterior probability ρ_j becomes directly proportional to the likelihood, $P(E_j \mid Z_j = \mathbb{P})$.

Personalized Fusion The Fusion Module leverages the posterior probability $\rho_j = P(Z_j = \mathbb{P} \mid E_j)$ to orchestrate a principled soft aggregation of parameters. The value of ρ_j is not merely a score but a direct, interpretable measure of confidence that parameter Φ_j should be personalized. Consequently, it can be used directly to govern the fusion process. The core principle is a parameter-wise convex combination of the globally aggregated model and the locally trained model. The weight assigned to the locally trained parameter, $\Phi_{i,j}^{(t,K)}$, is its posterior probability of being $\mathbb{P}$, ρ_j. Conversely, the weight for the global model parameter, $\Phi_{\mathcal{G},j}^{(t+1)}$, is its probability of being $\mathbb{G}$, which is $P(Z_j = \mathbb{G} \mid E_j) = 1 - \rho_j$. This leads to the following update rule:

$$\Phi_{i,j}^{(t+1)} = (1 - \rho_j) \cdot \Phi_{\mathcal{G},j}^{(t+1)} + \rho_j \cdot \Phi_{i,j}^{(t,K)}, \quad (11)$$

This formulation provides a seamless transition between global consensus and local specialization. A parameter with a high ρ_j will be strongly preserved from its local training.

Convergence Guarantee Our theoretical analysis is grounded on the following assumptions, which are common in the analysis of FL algorithms (Li et al. 2020b,a).

Assumption 1 (Smoothness): Each local objective $\mathcal{L}_i(w)$ is L–smooth, i.e., for any $w_1, w_2 \in \mathbb{R}^d$,

$$\mathcal{L}_i(w_1) \leq \mathcal{L}_i(w_2) + \langle \nabla \mathcal{L}_i(w_2), w_1 - w_2 \rangle + \frac{L}{2}\|w_1 - w_2\|^2, \quad (12)$$

Assumption 2 (Unbiased & Bounded Variance of Stochastic Gradients): For all clients i and parameters w,

$$\mathbb{E}[\nabla \tilde{L}_i(w)] = \nabla \mathcal{L}_i(w), \quad \mathbb{E}\|\nabla \tilde{L}_i(w) - \nabla \mathcal{L}_i(w)\|^2 \leq \sigma^2, \quad (13)$$

Assumption 3 (Gradient Diversity): The variation of local gradients across clients is bounded by a constant δ,

$$\frac{1}{N}\sum_{i=1}^{N} \|\nabla \mathcal{L}_i(w) - \nabla \mathcal{F}(w)\|^2 \leq \delta^2, \quad (14)$$

Assumption 4 (Bounded Personalization Score): The parameter–wise personalization weights $\rho_{n,j}$ inferred from behavioral evidence satisfy,

$$0 < \underline{\rho} \leq \rho_{n,j} \leq \overline{\rho} \leq 1, \quad (15)$$

and their variance is bounded, $\mathrm{Var}(\rho_{n,j}) \leq \sigma_\rho^2$.

Theorem 1 (Convergence of PPFL): Under Assumptions, we have the single-round t descent bound:

$$\mathbb{E}_t[\mathcal{F}(w_G^{(t+1)})] \leq \mathcal{F}(w_G^{(t)}) - \Big(\frac{\eta K \underline{\rho}}{2} - \frac{L\eta^2 K^2 \overline{\rho}^2}{2N}\Big) \cdot \|\nabla \mathcal{F}(w_G^{(t)})\|^2 + \Big(\frac{\eta K \overline{\rho}\, \delta^2}{4} + \frac{L\eta^2 K^2 \overline{\rho}^2 (\delta^2 + \sigma^2)}{2N}\Big), \quad (16)$$

Let we choose the learning rate $\eta = \frac{N\underline{\rho}}{2LK^2\overline{\rho}^2}$. After T communication rounds, the sequence $\{\Phi_G^{(t)}\}$ generated by the PPFL update satisfies:

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla \mathcal{F}(w_G^{(t)})\|^2 \leq \frac{8LK\overline{\rho}^2}{N\underline{\rho}^2 T}\left(\mathcal{F}(w_G^{(0)}) - \mathcal{F}^*\right) + \frac{\overline{\rho}}{\underline{\rho}}(\delta^2 + \sigma^2 + \sigma_\rho^2), \quad (17)$$

Remark: The bound (17) reveals two convergence phases: (*i*) the optimization term $\frac{8LK\overline{\rho}^2}{N\underline{\rho}^2 T}\Delta_0$ decays as $\mathcal{O}(1/T)$, and (*ii*) the stationary bias term $\frac{\overline{\rho}}{\underline{\rho}}(\delta^2 + \sigma^2 + \sigma_\rho^2)$ reflects intrinsic heterogeneity, stochastic noise, and uncertainty in the behavior–posterior inference. The proof sketch is as follows, with the full derivation provided in the Appendix.

Experiments

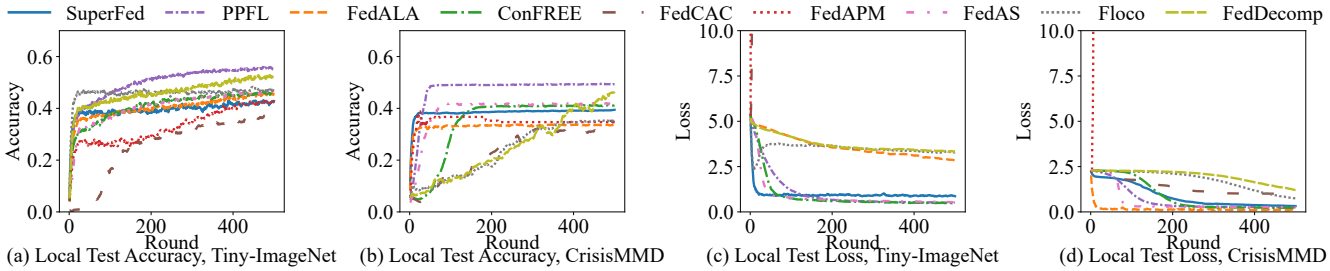
Experimental Setup

Datasets and Models We evaluate our method on four datasets: CIFAR100(CIFAR*) (Krizhevsky, Hinton et al. 2009) and Tiny-ImageNet(Tiny-Ig*) (Chrabaszcz, Loshchilov, and Hutter 2017) for image classification, AG News (Zhang, Zhao, and LeCun 2015) for text classification, and CrisisMMD (Alam, Ofli, and Imran 2018) for multimodal classification. We adopt Resnet18 (He et al. 2016), TextClassificationModel, and ImageTextClassificationModel accordingly, with their architectural details provided in the Appendix.

Baseline Methods To evaluate PPFL, we compare against state-of-the-art methods including the layer-decoupled FedAS (Yang, Huang, and Ye 2024), interpolation-based SuperFed (Hahn, Jeong, and Lee 2022) and Floco (Grinwald, Wiesner, and Nakajima 2024), aggregation-based FedALA (Zhang et al. 2023b), FedCAC (Wu et al. 2023), and ConFREE (Zheng et al. 2025), the drift-correction FedAPM (Zhu et al. 2025), and parameter-wise decomposition into

Settings	Pathological heterogeneous setting				Practical heterogeneous setting			
Datasets	CIFAR*	Tiny-Ig*	AGNEWS	CrisisMMD	CIFAR*	Tiny-Ig*	AGNEWS	CrisisMMD
FedAS	69.33(.21)	52.15(.33)	88.78(.18)	47.02(.45)	59.81(.19)	47.24(.12)	81.92(.22)	42.11(.38)
SuperFed	66.18(.46)	52.89(.51)	85.22(.39)	48.89(.62)	55.72(.31)	43.18(.25)	79.80(.34)	39.45(.41)
Floco	69.81(.15)	50.02(.28)	85.58(.16)	44.67(.31)	57.05(.11)	48.29(.14)	83.15(.19)	35.33(.29)
FedALA	70.11(.25)	56.39(.31)	84.18(.21)	42.15(.39)	58.99(.23)	46.13(.28)	82.53(.25)	33.78(.33)
FedCAC	52.55(.68)	49.14(.73)	72.43(.55)	46.40(.81)	31.33(.42)	38.06(.59)	77.21(.48)	35.19(.67)
ConFREE*	71.92(.11)	55.28(.19)	89.33(.13)	48.98(.25)	62.16(.09)	46.42(.11)	84.68(.15)	41.28(.22)
FedAPM	69.03(.18)	51.77(.24)	90.12(.17)	46.83(.28)	61.48(.14)	43.37(.18)	83.94(.20)	38.91(.26)
FedDecomp	72.55(.13)	60.10(.18)	89.21(.14)	52.53(.22)	62.24(.10)	52.89(.13)	87.85(.14)	46.46(.20)
PPFL	**75.26(.10)**	**65.43(.15)**	**91.94(.11)**	**54.23(.20)**	**64.15(.08)**	**55.98(.10)**	**88.22(.12)**	**49.54(.18)**

Table 1: The local test accuracy (%) in the pathological heterogeneous setting and practical heterogeneous setting with K=8 local update steps. Data are presented as mean(SD). ConFREE* denotes FedPAC+ConFREE.



(a) Local Test Accuracy, Tiny-ImageNet (b) Local Test Accuracy, CrisisMMD (c) Local Test Loss, Tiny-ImageNet (d) Local Test Loss, CrisisMMD

Figure 3: Comparison of accracy and loss curve across various methods.

personalized and global components FedDecomp (Wu et al. 2024). We also present plugin-enhanced versions of some TFL algorithms (Li et al. 2020a; Wang et al. 2020; Karimireddy et al. 2020; Li, He, and Song 2021).

Implementation Details Unless explicitly specified, we simulate a federated learning setup with 100 clients and 20% participation per round. We use a batch size of 64, learning rate of 0.01, and train for 500 rounds. Following prior work (Zhang et al. 2023b,a,c), we consider both pathological and practical data heterogeneity. Each client's local data split 3:1 into training and test sets. We run each experiment with 3 random seeds and report the average and standard deviation.

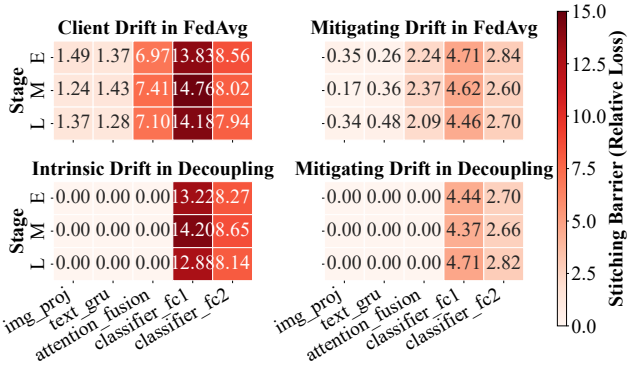


Figure 4: Mitigating Client Drift and Intrinsic Drift.

Performance and Cost Analysis

Accuracy Comparison As shown in Table 1 and Fig. 3, we compare PPFL with SOTA baselines on image, text, and multimodal classification tasks. In a practical heterogeneous setting, PPFL outperforms the second-ranked FedDecomp by 3.09% and the third-ranked Floco by 7.69% on Tiny-ImageNet. This is due to PPFL's parameter-level aggregation, distinguishing it from layer-level methods. FedALA, which applies parameter-level aggregation only to the final layer, performs worse than both PPFL and FedDecomp. FedCAC, despite using parameter-level aggregation, suffers from significant aggregation noise, hindering convergence accuracy. In a pathological heterogeneous setting, PPFL surpasses the suboptimal FedDecomp by 5.3%, showing its robustness in handling irrelevant information through local aggregation based on update trends.

Performance Gain over TFL Since PPFL does not modify the original FedAvg steps, it acts as a plug-in personalization engine. As shown in Table 3, integrating PPFL leads to significant accuracy improvements: 35.13% on AG NEWS and 16.82% on Tiny-ImageNet when combined with FedAvg. For SCAFFOLD, which corrects client drift, PPFL boosts accuracy by 14.07% on AG NEWS. Similarly, integrating PPFL into MOON, which aligns representations via contrastive learning, improves accuracy by 14.89% on Tiny-ImageNet. Unlike methods that constrain local training trajectories, PPFL intervenes post-training, treating w_{local} as the final product and focusing on inter-round fusion to create the next-state personalized model w_{next}. By decoupling

Ablation Study	Test Accuracy (%)
Effect of Local Iterations (K)	
K=1	52.45 ± 0.62
K=5	55.13 ± 0.48
K=10	55.82 ± 0.55
Num Clients N and Selection Ratio γ	
N=100, $\gamma = 0.2$	54.21 ± 0.51
N=100, $\gamma = 1.0$	55.13 ± 0.48
N=500, $\gamma = 0.2$	57.88 ± 0.73
N=500, $\gamma = 1.0$	58.45 ± 0.65
N=1000, $\gamma = 0.2$	59.74 ± 0.89
N=1000, $\gamma = 1.0$	61.02 ± 0.77
Effect of Core Modules	
PPFL (Full Modules)	55.13 ± 0.48
w/o PBA. (Magnitude-only)	49.59 ± 0.67
w/o PF. (Avg. Agg.)	51.24 ± 0.55

Table 2: Comprehensive Ablation Study on the Tiny-Ig* with Dir α=0.1. PBA. and PF. are abbreviations for parameter behavior awareness and personalized fusion.

the client's global contribution from personalized model refinement, PPFL seamlessly integrates with server-side protocols, enabling easy upgrades to existing federated deployments without altering the underlying infrastructure.

Overhead Analysis As shown in Table 4, we report the computational and communication overhead. PPFL is the most computationally efficient among all baselines, outperforming the second-most efficient method, ConFREE, by 9.0%, and accelerating computation by 3.25× compared to Floco. This efficiency arises from PPFL's lightweight, client-only design, where the perception and fusion modules use optimized, parameter-free computations. In terms of communication, PPFL adds no extra burden.

Ablation Study

As shown in Table 2, we conduct a comprehensive ablation study of PPFL. ***Effect of Local Iterations***: As K increases (batch size reduced to maintain total data per round), performance improves from 52.45% at K=1 to 55.82% at K=10, due to the perception module capturing stable, dynamic signals at larger K. At K=1, noisy updates rely on less informative static evidence, reducing accuracy. ***Scalability***: With 1000 clients and 20% participation, PPFL achieves 59.74% accuracy, demonstrating its robustness across varying client counts and participation rates. ***Parameter Behavior***: Relying solely on update magnitude reduces performance by 5.54%, highlighting the importance of integrating multiple behavioral signals to accurately identify a stable personalized core. ***Fusion***: Using simple model averaging drops accuracy by 3.89%, emphasizing the role of intelligent fusion, guided by the perception module's posterior.

Visualization of Drift Reduction

Fig. 4 quantifies the loss barriers arising from both client and intrinsic drift. The top row shows client drift in FedAvg, with a peak barrier of 14.76 in the classifier layer. The top-right subplot shows the effect of the PPFL fusion mechanism, which significantly reduces drift across all layers, lowering the maximum barrier from 14.76 to 4.62. The bottom row visualizes intrinsic drift in a partially decoupled PFL method, replacing the personalized head and resulting in a large barrier of 14.20, indicating functional decoupling. In contrast, the bottom-right subplot shows PPFL's fusion mechanism, which intelligently merges the old and new models, reducing the intrinsic drift barrier from 14.20 to 4.37.

Methods	Heterogeneous setting $\alpha = 3$		
Datasets	Tiny-Ig*	AG NEWS	CrisisMMD
FedAvg	19.35	44.58	27.21
+ PPFL	**+16.82**↑	**+35.13**↑	**+7.55**↑
FedProx	19.04	43.92	26.88
+ PPFL	**+12.17**↑	**+24.73**↑	**+7.02**↑
FedNova	23.28	46.55	29.94
+ PPFL	**+13.27**↑	**+26.93**↑	**+5.94**↑
MOON	20.12	46.03	32.78
+ PPFL	**+14.89**↑	**+27.86**↑	**+5.37**↑
SCAFFOLD	17.81	42.15	26.53
+ PPFL	**+10.32**↑	**+14.07**↑	**+4.31**↑

Table 3: Serving as a personalization-enhancing plug-in to improve the final local accuracy (%) of TFL algorithms.

	Comp.		Comm.
Methods	Total time	Time/rnd.	Param./rnd.
FedAvg + Finetuning	2057 min	246 s	2 * Σ
FedAS	1712 min	198 s	2 * Σ
SuPerFed	1461 min	174 s	2 * Σ
Floco	3155 min	372 s	2 * Σ + 2 * (M_{v} - 1) * Σ_{last}
FedALA	1403 min	156 s	2 * Σ
FedCAC	1236 min	147 s	$\sim$ 3 * Σ
ConFREE	1067 min	126 s	2 * Σ
FedAPM	1559 min	186 s	2 * Σ
FedDecomp	1365 min	162 s	2 * Σ
PPFL	**971 min**	**114 s**	**2 *** Σ

Table 4: Overhead analysis of different methods in the Tiny-ImageNet (100 clients, 500 rounds) scenario. Σ represents the model's parameter size, M_{v} is the number of simplex vertices, and Σ_{last} denotes the parameters of the final layer.

Conclusion

In this paper, we present PPFL, a dynamic method that combines parameter behavior for PFL. It can serve as a plug-in, effectively enhancing TFL methods. It outperforms eight SOTA methods across image, text, and multimodal tasks.

Acknowledgements

This research was supported by Youth Innovation Fund of the School of Software Engineering, University of Science and Technology of China (YN2260080008).

References

Adilova, L.; Andriushchenko, M.; Kamp, M.; Fischer, A.; and Jaggi, M. 2024. Layer-wise linear mode connectivity. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Alam, F.; Ofli, F.; and Imran, M. 2018. CrisisMMD: Multimodal Twitter Datasets from Natural Disasters. In *Proceedings of the Twelfth International Conference on Web and Social Media, ICWSM 2018, Stanford, California, USA, June 25-28, 2018*, 465–473. AAAI Press.

Arivazhagan, M. G.; Aggarwal, V.; Singh, A. K.; and Choudhary, S. 2019. Federated Learning with Personalization Layers. *CoRR*, abs/1912.00818.

Bengio, Y. 2012. Deep Learning of Representations for Unsupervised and Transfer Learning. In Guyon, I.; Dror, G.; Lemaire, V.; Taylor, G. W.; and Silver, D. L., eds., *Unsupervised and Transfer Learning - Workshop held at ICML 2011, Bellevue, Washington, USA, July 2, 2011*, volume 27 of *JMLR Proceedings*, 17–36. JMLR.org.

Chen, H.; and Chao, W. 2022. On Bridging Generic and Personalized Federated Learning for Image Classification. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Chrabaszcz, P.; Loshchilov, I.; and Hutter, F. 2017. A Downsampled Variant of ImageNet as an Alternative to the CIFAR datasets. *CoRR*, abs/1707.08819.

Collins, L.; Hassani, H.; Mokhtari, A.; and Shakkottai, S. 2021. Exploiting Shared Representations for Personalized Federated Learning. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, 2089–2099. PMLR.

Deng, Y.; Kamani, M. M.; and Mahdavi, M. 2020. Adaptive Personalized Federated Learning. *CoRR*, abs/2003.13461.

Grinwald, D.; Wiesner, P.; and Nakajima, S. 2024. Federated Learning over Connected Modes. In Globersons, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J. M.; and Zhang, C., eds., *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Hahn, S.; Jeong, M.; and Lee, J. 2022. Connecting Low-Loss Subspace for Personalized Federated Learning. In Zhang, A.; and Rangwala, H., eds., *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, 505–515. ACM.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 770–778. IEEE Computer Society.

Hsieh, K.; Phanishayee, A.; Mutlu, O.; and Gibbons, P. B. 2020. The Non-IID Data Quagmire of Decentralized Machine Learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, 4387–4398. PMLR.

Karimireddy, S. P.; Kale, S.; Mohri, M.; Reddi, S. J.; Stich, S. U.; and Suresh, A. T. 2020. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, 5132–5143. PMLR.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Li, Q.; He, B.; and Song, D. 2021. Model-Contrastive Federated Learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, 10713–10722. Computer Vision Foundation / IEEE.

Li, Q.; Wen, Z.; Wu, Z.; Hu, S.; Wang, N.; Li, Y.; Liu, X.; and He, B. 2023. A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. *IEEE Trans. Knowl. Data Eng.*, 35(4): 3347–3366.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020a. Federated Optimization in Heterogeneous Networks. In Dhillon, I. S.; Papailiopoulos, D. S.; and Sze, V., eds., *Proceedings of the Third Conference on Machine Learning and Systems, MLSys 2020, Austin, TX, USA, March 2-4, 2020*. mlsys.org.

Li, X.; Huang, K.; Yang, W.; Wang, S.; and Zhang, Z. 2020b. On the Convergence of FedAvg on Non-IID Data. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Luo, J.; and Wu, S. 2022. Adapt to Adaptation: Learning Personalization for Cross-Silo Federated Learning. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, 2166–2173. ijcai.org.

Maini, P.; Mozer, M. C.; Sedghi, H.; Lipton, Z. C.; Kolter, J. Z.; and Zhang, C. 2023. Can Neural Network Memorization Be Localized? In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, 23536–23557. PMLR.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Singh, A.; and Zhu, X. J., eds., *Proceedings of the 20th International*

Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA, volume 54 of *Proceedings of Machine Learning Research*, 1273–1282. PMLR.

Oh, J.; Kim, S.; and Yun, S. 2021. FedBABU: Towards Enhanced Representation for Federated Image Classification. *CoRR*, abs/2106.06042.

Raghu, M.; Gilmer, J.; Yosinski, J.; and Sohl-Dickstein, J. 2017. SVCCA: Singular Vector Canonical Correlation Analysis for Deep Learning Dynamics and Interpretability. In Guyon, I.; von Luxburg, U.; Bengio, S.; Wallach, H. M.; Fergus, R.; Vishwanathan, S. V. N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 6076–6085.

Tan, A. Z.; Yu, H.; Cui, L.; and Yang, Q. 2023. Towards Personalized Federated Learning. *IEEE Trans. Neural Networks Learn. Syst.*, 34(12): 9587–9603.

Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; and Poor, H. V. 2020. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Wang, Z.; Fan, X.; Qi, J.; Wen, C.; Wang, C.; and Yu, R. 2021. Federated Learning with Fair Averaging. In Zhou, Z., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, 1615–1623. ijcai.org.

Wu, X.; Liu, X.; Niu, J.; Wang, H.; Tang, S.; Zhu, G.; and Su, H. 2024. Decoupling General and Personalized Knowledge in Federated Learning via Additive and Low-rank Decomposition. In Cai, J.; Kankanhalli, M. S.; Prabhakaran, B.; Boll, S.; Subramanian, R.; Zheng, L.; Singh, V. K.; César, P.; Xie, L.; and Xu, D., eds., *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, 7172–7181. ACM.

Wu, X.; Liu, X.; Niu, J.; Zhu, G.; and Tang, S. 2023. Bold but Cautious: Unlocking the Potential of Personalized Federated Learning through Cautiously Aggressive Collaboration. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 19318–19327. IEEE.

Yang, X.; Huang, W.; and Ye, M. 2024. FedAS: Bridging Inconsistency in Personalized Federated Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, 11986–11995. IEEE.

Yi, L.; Wang, G.; Liu, X.; Shi, Z.; and Yu, H. 2023. FedGH: Heterogeneous Federated Learning with Generalized Global Header. In El-Saddik, A.; Mei, T.; Cucchiara, R.; Bertini, M.; Vallejo, D. P. T.; Atrey, P. K.; and Hossain, M. S., eds., *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023- 3 November 2023*, 8686–8696. ACM.

Zhang, J.; Hua, Y.; Wang, H.; Song, T.; Xue, Z.; Ma, R.; Cao, J.; and Guan, H. 2023a. GPFL: Simultaneously Learning Global and Personalized Feature Information for Personalized Federated Learning. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 5018–5028. IEEE.

Zhang, J.; Hua, Y.; Wang, H.; Song, T.; Xue, Z.; Ma, R.; and Guan, H. 2023b. FedALA: Adaptive Local Aggregation for Personalized Federated Learning. In Williams, B.; Chen, Y.; and Neville, J., eds., *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, 11237–11244. AAAI Press.

Zhang, J.; Hua, Y.; Wang, H.; Song, T.; Xue, Z.; Ma, R.; and Guan, H. 2023c. FedCP: Separating Feature Information for Personalized Federated Learning via Conditional Policy. In Singh, A. K.; Sun, Y.; Akoglu, L.; Gunopulos, D.; Yan, X.; Kumar, R.; Ozcan, F.; and Ye, J., eds., *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2023, Long Beach, CA, USA, August 6-10, 2023*, 3249–3261. ACM.

Zhang, J.; Liu, Y.; Hua, Y.; Wang, H.; Song, T.; Xue, Z.; Ma, R.; and Cao, J. 2023d. PFLlib: Personalized Federated Learning Algorithm Library. *CoRR*, abs/2312.04992.

Zhang, L.; Luo, Y.; Bai, Y.; Du, B.; and Duan, L. 2021a. Federated Learning for Non-IID Data via Unified Feature Learning and Optimization Objective Alignment. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, 4400–4408. IEEE.

Zhang, M.; Sapra, K.; Fidler, S.; Yeung, S.; and Álvarez, J. M. 2021b. Personalized Federated Learning with First Order Model Optimization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Zhang, X.; Zhao, J. J.; and LeCun, Y. 2015. Character-level Convolutional Networks for Text Classification. In Cortes, C.; Lawrence, N. D.; Lee, D. D.; Sugiyama, M.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, 649–657.

Zheng, H.; Hu, Z.; Yang, L.; Zheng, M.; Xu, A.; and Wang, B. 2025. ConFREE: Conflict-free Client Update Aggregation for Personalized Federated Learning. In Walsh, T.; Shah, J.; and Kolter, Z., eds., *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, 22875–22883. AAAI Press.

Zhu, S.; Nie, F.; Zeng, J.; Wang, S.; Sun, Y.; Yao, Y.; Chen, S.; Xu, Q.; and Yang, C. 2025. FedAPM: Federated Learning via ADMM with Partial Model Personalization. *CoRR*, abs/2506.04672.