HEART-PFL: Stable Personalized Federated Learning under Heterogeneity with Hierarchical Directional Alignment and Adversarial Knowledge Transfer

Minjun Kim*
Promedius Inc.
weightboy7@gmail.com

Minje Kim*,†
Promedius Inc.
iankimrok@gmail.com

Abstract

Personalized Federated Learning (PFL) aims to deliver effective client-specific models under heterogeneous distributions, yet existing methods suffer from shallow prototype alignment and brittle server-side distillation. We propose HEART-PFL, a dual-sided framework that (i) performs depth-aware Hierarchical Directional Alignment (HDA) using cosine similarity in the early stage and MSE matching in the deep stage to preserve client specificity, and (ii) stabilizes global updates through Adversarial Knowledge Transfer (AKT) with symmetric KL distillation on clean and adversarial proxy data. Using lightweight adapters with only 1.46M trainable parameters, HEART-PFL achieves state-of-the-art personalized accuracy on CIFAR-100, Flowers-102, and Caltech-101 (63.42%, 84.23%, and 95.67%, respectively) under Dirichlet non-IID partitions, and remains robust to out-of-domain proxy data. Ablation studies further confirm that HDA and AKT provide complementary gains in alignment, robustness, and optimization stability, offering insights into how the two components mutually reinforce effective personalization. Overall, these results demonstrate that HEART-PFL simultaneously enhances personalization and global stability, highlighting its potential as a strong and scalable solution for PFL (code available at https://github.com/danny0628/HEART-PFL).

1. Introduction

Federated learning (FL) is a distributed learning framework that enables multiple clients to collaboratively train models without requiring centralized data collection, thereby preserving data privacy and supporting deployment across diverse domains such as healthcare and computer vision. As a fundamental solution, the FedAvg [28] algorithm updates a global model on the server by aggregating parameters from locally trained clients. One of the central challenges in FL arises from client heterogeneity, which commonly manifests as imbalanced class distributions or non-independent and identically distributed (non-IID) data [4, 6, 20, 41]. In response to personalization under client heterogeneity, Personalized Federated Learning (PFL) [35] couples client-specific adaptation with server-mediated knowledge aggregation to improve per-client performance.

Previous studies have investigated diverse personalization strategies. Early approaches adopted full-model personalization [8, 9, 19, 27], in which each client optimizes a personalized model with a regularization term while the global model resides on the server. This strategy is parameter-intensive and imposes substantial computation and communication overhead, limiting its practicality for resource-constrained clients. To alleviate the overhead, subsequent works proposed partial-model personalization [1, 3, 30, 46], where each client splits its model into personalized and shared components, exchanging only shared parameters with the server.

Adapter-based personalization [32] reduces computation by personalizing only lightweight adapter modules while sharing a common backbone across clients. PerAda [40] extends this idea by freezing all non-adapter parameters and updating only the adapter modules, enabling highly efficient personalization.

Prototype-based approaches [7, 25, 38] typically perform alignment using prototypes from a single layer and a predominantly server→client flow, leaving hierarchical semantics underutilized and limiting client specificity. FedPHP [21], for example, depends on global prototypes whose quality mirrors that of personalized extractors—a fragile circular dependency. Meanwhile, GPFL [44] introduces trainable category embeddings that guide features in both magnitude and direction without relying on strong extractors.

Knowledge transfer among clients via the server can guide the training process and improve personalization [43]. Incorporating clients' historical personalized knowledge [11] further enhances local adaptation. How-

*Equal contribution
†Corresponding author

ever, existing knowledge distillation approaches often improve personalization at the expense of degrading the global model. To mitigate this trade-off, PerAda distills clients' ensemble knowledge into the global model, improving generalization without sacrificing personalization.

Motivation and Observation. Despite recent progress, PFL still faces two persistent gaps under non-IID client distributions and out-of-domain data. First, representation alignment is often shallow and one-way: prototype-based strategies typically align at a only single layer and mostly in the server-to-client direction, underutilizing hierarchical semantics and risking the suppression of client-specific cues essential for personalization. Second, server-side knowledge transfer remains brittle: ensemble distillation on in-distribution unseen data tends to push the global model toward an averaged client teacher in a one-sided manner, which can propagate teacher bias, fail to convey representational capacity, and destabilize global updates under heterogeneity.

Our Approach. We present HEART-PFL, which directly targets these two challenges with two complementary components that constitute our main contribution. Hierarchical Directional Alignment (HDA) performs layer-wise, direction-aware alignment between global features and client class prototypes: cosine-based directional agreement in early stage encourages consistent geometry for generic patterns, while MSE-based semantic matching in deep stage preserves class-specific meaning—leveraging the full hierarchy without erasing personalization (Fig. 1). Adversarial Knowledge Transfer (AKT) strengthens server-side transfer by making distillation bidirectional and robust. The global adapter and the client ensemble are aligned in both directions on clean and adversarial proxy views, increasing feature diversity and stabilizing updates despite client heterogeneity (Fig. 2). For computational and communication efficiency, we adopt an adapter-based realization following prior work [32, 40]; this is a supporting design choice rather than our core contribution.

Empirical evidence. We evaluate HEART-PFL under standard PFL regime with label-skew, non-IID clients generated via Dirichlet sampling (various α) and with varying client participation rates. Across these heterogeneous settings, HEART-PFL consistently yields higher client personalization on the all benchmark datasets. Specifically, the ablation studies highlight the complementary roles of the two modules: HDA yields hierarchy-aware alignment improvements, AKT enhances robustness and server-side stability, and their combination achieves the largest personalization gains with the most stable training. We summarize our main contributions below:

- We propose the HEART-PFL framework, which tackles the data heterogeneity inherent in PFL through dual-side contributions: client-side alignment and server-side knowledge transfer, resulting in enhanced personalization effectiveness.
- Our proposed method has two key components: (i) HDA, which aligns global features with personalized prototypes across multiple network layers; and (ii) AKT, which enhances the robustness of knowledge transfer via adversarial views and symmetric distillation.
- We conduct extensive evaluations on several benchmark datasets with heterogeneous settings. Our method demonstrates superior performance compared to state-of-the-art methods across diverse heterogeneous client distributions. Moreover, we further demonstrated the effectiveness of our key components, HDA and AKT, through comprehensive ablation studies.

2. Related Works

2.1. Federated Learning under Data Heterogeneity

FL often suffers significant degradation under non-IID or class-imbalanced client data, which induces client drift and destabilizes global aggregation. The canonical FedAvg [28] struggles in such regimes, and subsequent approaches have introduced proximal constraints or variance-reduction corrections to mitigate client drift and improve convergence on skewed data [12, 18]. A complementary line of work focuses on aligning client representations via class prototypes. Methods such as FedProto [38] and its successors [7, 25] promote feature–prototype agreement to enhance consistency and stabilize knowledge sharing amid heterogeneous client distributions. Nonetheless, existing prototype formulations typically perform alignment at only a single feature layer and rely on a predominantly server-to-client alignment flow, which in turn underutilizes hierarchical semantic structure and attenuates client-specific characteristics.

Knowledge distillation approaches address robustness from a different angle. Server-side distillation from an ensemble of client teachers has been shown to improve the global model's resilience to heterogeneous inputs [23, 24]. However, unilateral distillation on clean proxy data often propagates teacher bias and can destabilize global updates when clients diverge. More recent variants incorporate refined distillation strategies, yet existing approaches [15, 42] in FL still restrict knowledge transfer to a one-way server-to-client direction. Complementary representation learning methods, including contrastive approaches [2, 16], aim to improve robustness under heterogeneity but typically operate at a single representation level and do not exploit the hierarchical structure available in deep networks. These limitations highlight the need for FL methods that can leverage multi-layer semantic information while ensuring robustness through bidirectional knowledge transfer.

2.2. Personalized Federated Learning

PFL aims to maximize per-client utility without forfeiting cross-client knowledge sharing. Early PFL approaches personalize the full model while regularizing toward a global reference (e.g., pFedMe [36], Ditto [19]), which improves specialization but incurs substantial computational and communication overhead. Parameter-splitting methods alleviate this burden by separating shared and private parameter subsets (e.g., FedPer [1], LG-FedAvg [22]), and recent systems refine this idea through selective or adaptive update schedules (e.g., FedSelect [37], FedALA [45]). Adapter-based designs further reduce overhead by training compact, client-specific modules atop a shared backbone (e.g., PerAda [40]).

Prototype-driven PFL aligns global features with client-specific prototypes, but its performance often depends heavily on extractor quality. Moreover, most formulations restrict alignment to a single representation layer with a predominantly server-to-client flow (e.g., FedPHP [21], FedProto [38]). More recent efforts (e.g., FedGPD [25], FedORGP [7]) advance this line but still center on shallow, unidirectional prototype alignment. Distillation-centric PFL, in contrast, transfers personalized knowledge through the server or historical teachers. Although modern formulations explore balanced divergence [33] and revised KL strategies [39], most PFL applications still rely on conventional teacher–student pipelines without incorporating adversarial perturbations or symmetric constraints.

Within this landscape, our method integrates two complementary components: (i) HDA between global features and personalized class prototypes to capture multi-layer semantics while preserving client-specific information, and (ii) bidirectional, symmetric knowledge distillation on both clean and adversarial proxy views to stabilize global updates under heterogeneity. Together, these components yield improvements in both personalized and global performance across strong PFL baselines while reducing round-to-round instability.

3. Methodology

3.1. Preliminaries

Personalized Federated Learning. We consider a traditional PFL setting with N clients $C = \{c_1, \ldots, c_N\}$, where each client c_k has a local non-IID dataset $\mathcal{D}_k = \{(\mathbf{x}_k^i, \mathbf{y}_k^i)\}_{i=1}^{n_k}$ containing n_k data samples. Let $\theta \in \mathbb{R}^{d_\theta}$ denote the parameters of global model shared across all clients, and $\Omega = \{\omega_1, \omega_2, \ldots, \omega_N\}$ represent the collection of parameters of personalized models, where each $\omega_k \in \mathbb{R}^{d_\omega}$ is specific to client c_k. The typical objective of PFL is formulated as:

$$\min_{\theta,\Omega} \mathcal{P}(\theta, \Omega) = \frac{1}{N} \sum_{k=1}^{N} \mathcal{F}_k(\theta, \omega_k), \tag{1}$$

where $\mathcal{F}_k(\theta, \omega_k)$ denotes the local objective function for client c_k:

$$\mathcal{F}_k(\theta, \omega_k) = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathcal{L}(f(\mathbf{x}_k^i; \theta, \omega_k), \mathbf{y}_k^i), \tag{2}$$

where $f(\cdot; \theta, \omega_k)$ is the model parameterized by both global and personalized parameters, and $\mathcal{L}(\cdot, \cdot)$ is the task-specific loss function.

Adapter for cost efficiency. Mitigating communication and computational overhead remains a critical challenge in PFL. To address this, we adopt a lightweight adapter architecture [40] implemented as a sequence of BatchNorm–Dropout–Conv2D layers with a residual connection, leveraging frozen pretrained parameters to maximize efficiency. Unlike conventional formulations where both global and personalized models are fully trainable, our framework restricts trainable components to adapter modules.

Formally, we redefine the global model as $\theta = (\psi, \tilde{\theta})$ and the personalized model as $\omega_k = (\psi, \tilde{\omega}_k)$, where ψ denotes frozen pretrained parameters and $\tilde{\theta}, \tilde{\omega}_k$ are the trainable adapter parameters. The personalized objective is thus reformulated as:

$$\min_{\tilde{\theta},\tilde{\Omega}} \mathcal{P}(\tilde{\theta}, \tilde{\Omega}) = \frac{1}{N} \sum_{k=1}^{N} \mathcal{F}_k(\tilde{\theta}, \tilde{\omega}_k). \tag{3}$$

3.2. Hierarchical Directional Alignment

Existing prototype-based approaches [7, 25, 38] have focused primarily on single-layer representations, which may not fully leverage the hierarchical semantic structure inherent in deep networks. Although these methods have shown promising results, they typically extract prototypes from only a single network layer, potentially overlooking information from the rest of the layers. Moreover, these methods predominantly follow a server-to-client alignment direction, where client models are regularized to conform to global prototypes. This is often at the expense of preserving the unique features required for effective personalization.

To address these challenges, we introduce HDA (Fig. 1), a novel layer-wise alignment mechanism that utilizes features of the global model θ with class-wise prototypes of the personalized model ω_k. As illustrated in Figure 1, HDA captures structural relationships between global and client parameter spaces via layer-specific alignment strategies, thereby enhancing personalized performance for all clients.

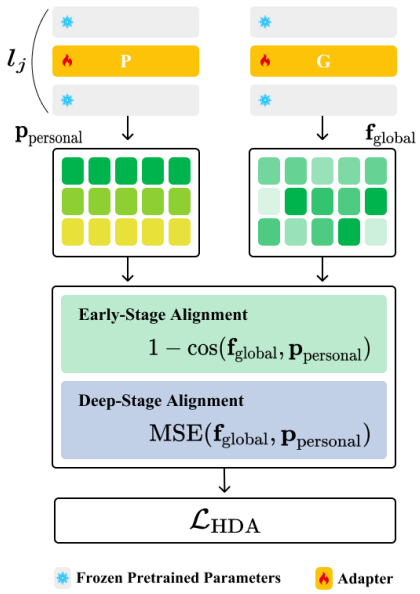


Figure 1. Overview of our proposed HDA. HDA extracts a hierarchical set of class-wise prototypes from each client's personalized model. It then aligns features from the global model with these client-specific prototypes using a semantic-aware mechanism: cosine similarity for early stage and MSE for deep stage. This alignment mechanism is formulated as our proposed HDA loss, $\mathcal{L}_{\text{HDA}}$.

Layer-wise Prototype Extraction. For each client c_k, we extract multi-scale feature representations across l network layers, spanning from elementary feature to semantic features. Given the client dataset $\mathcal{D}_k$, we formalize the layer-wise feature extraction process as:

$$\mathbf{F}_k^{(l)} = \{h_k^{(l)}(\mathbf{x}_i) : (\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_k\}, \tag{4}$$

where h denotes the feature extraction function, followed by adaptive pooling to ensure dimensional consistency.

Class-wise Prototype Extraction. For each class v and layer l, we construct personalized prototypes by aggregating class-representative features:

$$\mathbf{p}_{k,v}^{(l)} = \frac{1}{|\mathcal{D}_{k,v}|} \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}_{k,v}} h_k^{(l)}(\mathbf{x}_i). \tag{5}$$

This generates a hierarchical prototype collection $\mathbf{P}_k$, capturing abundant class representations for client k. The class-wise prototype extraction is consistent across heterogeneous data distributions, aiding reliable convergence within each client's imbalanced class distribution.

Semantic-Aware Layer-wise Alignment. The core innovation of HDA lies in its semantic-aware alignment mechanism that employs stage-wise similarity functions tailored to different representational characteristics. The semantic-aware alignment function is defined as:

$$\mathcal{A}^{(l)} = \begin{cases} 1 - \cos(\mathbf{f}_{\text{global}}, \mathbf{p}_{\text{personal}}), & l \in l_{\text{early}}, \\ \text{MSE}(\mathbf{f}_{\text{global}}, \mathbf{p}_{\text{personal}}), & l \in l_{\text{deep}}. \end{cases} \tag{6}$$

Our asymmetric design leverages global features $\mathbf{f}_{\text{global}}$ extracted by the global model and aligns them with client-specific, class-wise prototypes $\mathbf{p}_{\text{personal}}$, effectively bridging client-specific understanding with collective global knowledge. Here, l_{early} and l_{deep} denote the layer sets of the early stage and the deep stage, respectively. We partition layers by depth: the early stage spans from the input to the network's central region, and the deep stage covers the layers beyond this boundary, as validated by our ablation study (Figure 4). In the early stage, where features are more generic, we employ cosine similarity to guide their directional alignment. For the deep stage, we use Euclidean distance to enforce precise alignment.

The complete training objective for each client combines the standard cross-entropy loss, proximal term $\frac{1}{2}\|\omega_k - \theta\|_2^2$, and our proposed HDA loss:

$$\mathcal{L}_{\text{HDA},k}(\theta, \mathbf{P}_k) = \frac{1}{n_k} \sum_{i=1}^{n_k} \sum_{j=1}^{l} \frac{1}{l} \mathcal{A}^{(j)}(h_g^{(j)}(\mathbf{x}_k^i; \theta), \mathbf{p}_{k,\mathbf{y}_k^i}^{(j)}). \tag{7}$$

Consequently, the final personalized objective function is:

$$\min_{\theta,\Omega} \mathcal{P}(\theta, \Omega) = \frac{1}{N} \sum_{k=1}^{N} \{\mathcal{F}_k(\theta, \omega_k) + \mathcal{L}_{\text{HDA},k}(\theta, \mathbf{P}_k)\}. \tag{8}$$

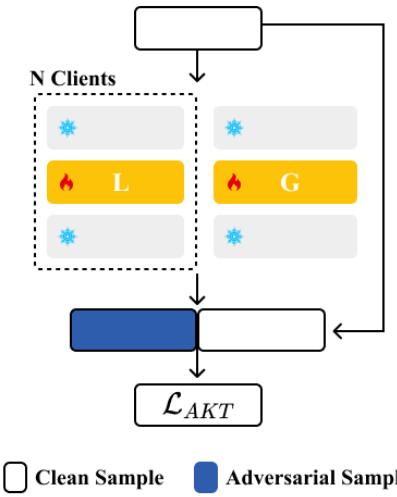


Figure 2. Overview of our proposed AKT. To enhance the robustness of the global adapter, AKT performs knowledge distillation using both clean and adversarially generated proxy samples.

3.3. Adversarial Knowledge Transfer

We propose an AKT mechanism that improves the generalization of the global adapter by enhancing the robustness of ensemble Knowledge Transfer (EKT) [23, 40] from heterogeneous clients to the server. Beyond aggregating client-specific parameters, our AKT further distills the client-ensemble knowledge to update the global adapter, thereby ensuring stable convergence under client heterogeneity.

On the server side, EKT employs a proxy dataset $\mathcal{D}_\rho$ to minimize the discrepancy between the probability distributions of the clients' local models and the global model:

$$\mathcal{L}_{EKT} = \sum_{\mathbf{x}_\rho \in \mathcal{D}_\rho} \mathcal{L}_{KL}(\bar{p}(\mathbf{x}_\rho) \,\|\, p_g(\mathbf{x}_\rho)), \tag{9}$$

where $\mathcal{L}_{KL}(\cdot \,\|\, \cdot)$ is the Kullback–Leibler divergence (KL) loss [10]. Here, $\bar{p}(\mathbf{x}_\rho) = \frac{1}{N}\sum_{i=1}^{N} p_i(\mathbf{x}_\rho)$ denotes the ensemble (averaged) client distribution, and $p_g(\mathbf{x}_\rho)$ denotes the probability distribution of the global model; both are obtained by applying the softmax to their corresponding logits.

However, this unilateral knowledge distillation from the local ensemble to the global model may lead to superficial transfer of knowledge. To address this, we enhance the EKT framework by incorporating adversarial knowledge and symmetric distillation, which promote feature diversity through adversarial samples and improve the stability of global adapter updates.

We utilize the symmetric KL loss, inspired by [14, 33, 39], to reduce representational discrepancies between the global adapter and the per-client local adapters under data heterogeneity. Specifically, for each client k, we instantiate a per-client local adapter $\tilde{\phi}_k$, used solely for EKT to the global adapter; $\tilde{\phi}_k$ mirrors the adapter architecture of both the personalized and global models. The symmetric KL acts as a regularizer, encouraging the global adapter to better capture the representational capacity of the local adapters while avoiding one-sided overfitting. Formally, the symmetric KL loss is defined as follows:

$$\begin{aligned}\mathcal{L}_{sKL} = &\sum_{\mathbf{x}_\rho \in \mathcal{D}_\rho} \mathcal{L}_{KL}\big(\bar{p}(\mathbf{x}_\rho) \,\|\, p_g(\mathbf{x}_\rho)\big) \\ &+ \sum_{\mathbf{x}_\rho \in \mathcal{D}_\rho} \mathcal{L}_{KL}\big(p_g(\mathbf{x}_\rho) \,\|\, \bar{p}(\mathbf{x}_\rho)\big).\end{aligned} \tag{10}$$

Then, we employ the standard adversarial perturbation method [26] to generate adversarial samples that embed robust knowledge. Given a clean sample $(\mathbf{x}_\rho, \mathbf{y}_\rho)$ from the proxy dataset $\mathcal{D}_\rho$, the adversarial counterpart $\mathbf{x}_\rho^{adv}$ is obtained via:

$$\mathbf{x}_\rho^{t+1} = \Pi_{\mathcal{B}_\epsilon(\mathbf{x}_\rho)}\Big(\mathbf{x}_\rho^t + \alpha \cdot \mathrm{sign}\big(\nabla_{\mathbf{x}_\rho^t} \mathcal{L}_{CE}(f(\mathbf{x}_\rho^t), \mathbf{y}_\rho)\big)\Big), \tag{11}$$

where $\mathcal{L}_{CE}(\cdot)$ is the cross-entropy loss, f is the model, α is the step size, and $\Pi_{\mathcal{B}_\epsilon(\mathbf{x}_\rho)}(\cdot)$ projects onto the ℓ_∞-ball of radius ϵ centered at $\mathbf{x}_\rho$. The process is initialized with $\mathbf{x}_\rho^0 = \mathbf{x}_\rho + \delta$, where $\delta \sim \mathrm{Unif}[-\epsilon, \epsilon]$. After T iterations, the final adversarial sample is $\mathbf{x}_\rho^{adv} = \mathbf{x}_\rho^T$.

Building upon the adversarial sample generation, we define the AKT loss as follows:

$$\mathcal{L}_{AKT} = \sum_{\mathbf{x}'_\rho \in \{\mathbf{x}_\rho, \mathbf{x}_\rho^{adv}\}} \mathcal{L}_{sKL}(\mathbf{x}'_\rho). \tag{12}$$

3.4. Algorithm

Algorithm 1 HEART-PFL

Input: Client set C, rounds R, Initialized models θ^0,$\{\tilde{\omega}_k^0\}_{k\in C}$, client datasets $\{\mathcal{D}_k\}_{k\in C}$, proxy dataset $\mathcal{D}_\rho$
Output: personalized adapters $\{\tilde{\omega}_k^R\}_{k\in C}$

for each round $r = \{0, 1, \ldots, R-1\}$ **do**
  $S_r \leftarrow$ Select a random subset of clients from C
  Receive global adapter $\tilde{\theta}^r$ from server
  // Client Side
  for client $c_k \in S_r$ **do**
    for epoch $e \in \{0, 1, \ldots, E_{\text{client}} - 1\}$ **do**
      Update $\tilde{\omega}_k^{r,e+1}$ by Eq. (8)
    $\tilde{\omega}_k^{r+1} \leftarrow \tilde{\omega}_k^{r,E_{\text{client}}}$
    Initialize local adapter $\tilde{\phi}_k^{r,0} \leftarrow \tilde{\theta}^r$
    for epoch $e \in \{0, 1, \ldots, E_{\text{client}} - 1\}$ **do**
      $\tilde{\phi}_k^{r,e+1} \leftarrow \tilde{\phi}_k^{r,e} - \eta_{\tilde{\phi}} \nabla \mathcal{L}_{CE}(\psi, \tilde{\phi}_k^{r,e})$
    Return $\tilde{\phi}_k^{r+1} = \tilde{\phi}_k^{r,E_{\text{client}}}$ to server
  // Server Side
  Adapter Averaging: $\tilde{\theta}^{r+1} \leftarrow \frac{1}{|S_r|}\sum_{c_k \in S_r} \tilde{\phi}_k^{r+1}$
  for each $(\mathbf{x}_\rho, \mathbf{y}_\rho) \in \mathcal{D}_\rho$ **do**
    $\mathbf{x}_{\rho,k}^{adv} \leftarrow$ generate by Eq. (11)$(\mathbf{x}_\rho;\, \phi_k^{r+1})$
    $\mathbf{x}_{\rho,g}^{adv} \leftarrow$ generate by Eq. (11)$(\mathbf{x}_\rho;\, \theta^{r+1})$
  for each $(\mathbf{x}_\rho, \mathbf{y}_\rho) \in \mathcal{D}_\rho$ **do**
    $\tilde{\theta}^{r+1} \leftarrow \tilde{\theta}^{r+1} - \eta_{\tilde{\theta}} \nabla \mathcal{L}_{AKT}(\mathbf{x}'_\rho)$
  Return updated global adapter $\tilde{\theta}^{r+1}$ to all clients

Algorithm 1 outlines the optimization of the personalized objective across rounds. At each communication round $r \in \{0, \ldots, R-1\}$, the server samples a client subset $S_r \in C$ and broadcasts the current global adapter $\tilde{\theta}^r$ to those clients. Each selected client $c_k \in S_r$ first updates its personalized adapter on local data $\mathcal{D}_k$ by minimizing the final personalized objective loss $\mathcal{P}(\cdot, \cdot)$ (Eq. 8) using HDA for client epochs $\tilde{\omega}_k^{r+1} \leftarrow \tilde{\omega}_k^{r,E_{\text{client}}}$. Then, the client initializes local adapter $\tilde{\phi}_k^{r,0} \leftarrow \tilde{\theta}^r$, and uses it with the standard cross-entropy loss $\mathcal{L}_{CE}(\psi, \tilde{\phi}_k)$ using learning rate $\eta_{\tilde{\phi}}$ over client epochs E_{client} to obtain $\tilde{\phi}_k^{r+1}$, which is sent back

to the server. After collecting $\tilde{\phi}_k^{r+1}$, the server performs simple adapter averaging, $\tilde{\theta}^{r+1} \leftarrow \frac{1}{|S_r|}\sum_{c_k \in S_r} \tilde{\phi}_k^{r+1}$. To strengthen $\tilde{\theta}^{r+1}$ against client heterogeneity via AKT, the server uses the proxy dataset $\mathcal{D}_\rho$ to generate adversarial variants via adversarial perturbation steps with respect to the local adapters and the current global adapter (Eq. 11), producing $\mathbf{x}_{\rho,k}^{\text{adv}}$ and $\mathbf{x}_{\rho,g}^{\text{adv}}$. Next, the server updates the global adapter on the combined samples $\mathbf{x}'_\rho$ by minimizing the AKT loss $\mathcal{L}_{\text{AKT}}$ with learning rate η_θ. Finally, the server broadcasts the updated adapter to initiate round $r+1$. After R rounds, HEART-PFL returns the personalized adapters $\{\tilde{\omega}_k^R\}_{k \in C}$.

4. Experimental Results

4.1. Experimental Setup

Datasets and Model. We conduct a classification task using three datasets: CIFAR100 [13], Flowers102 [29], and Caltech101 [5]. We perform experiments in a PFL environment with simulation settings. To simulate imbalanced data distributions across all three datasets, we create non-IID environments by employing Dirichlet distributions with various alpha parameters (0.1, 0.3, 0.5) across all clients. For the overall experiments, we adopt ResNet-18 pretrained on ImageNet-1K [34], following previous studies [6, 32, 37, 40].

Baselines. To evaluate our method, we compare against state-of-the-art PFL methods: LG-FedAvg [22], FedPer [1], Ditto [19], FedBABU [30], FedALA [45], and PerAda [40]. For a fair comparison in personalized experiments, we include FedAvg [28], FedProx [17], and FedProto [38], which are originally designed for generalized global model learning, diverging from PFL's personalization objectives. Furthermore, we evaluate personalized versions of these methods, derived by fine-tuning their global models. We denote them as FedAvg-Per, FedProx-Per, and FedProto-Per, respectively.

Implementation Details. For fair non-IID settings, we apply the following configuration across all participating clients per round and datasets. The total number of clients N is set to 20, with 8 clients randomly participating in each round. Each round consists of client epochs $E = 10$, and we conduct experiments for a total round $R = 200$. For client-side hyper-parameters, we use SGD optimizer with a learning rate of 0.01, learning rate decay of 1, and training batch size of 16. For server-side hyper-parameters in AKT, we employ Adam optimizer with a learning rate of 0.001 and batch size of 2048.

4.2. Personalization Performance Evaluation.

Table 1 summarizes the final experimental results, comparing our proposed method, HEART-PFL, with several state-of-the-art baselines across three public benchmark datasets. The experiments are conducted under various levels of data heterogeneity, simulated using a Dirichlet distribution. The results demonstrate that HEART-PFL consistently achieves the best performance across all datasets and under every tested Dirichlet setting. Specifically, when averaging the performance across all distributions, HEART-PFL demonstrates substantial improvements over various baseline categories. Compared to generic FL methods adapted for personalization, HEART-PFL achieves improvements of 16.56% on CIFAR100, 12.65% on Flowers102, and 3.38% on Caltech101. Against PFL methods with different approaches from HEART-PFL, our method shows gains of 11.14%, 12.83%, and 4.40% across the three datasets. When compared to feature alignment-based PFL methods, HEART-PFL outperforms by margins of 23.40%, 18.25%, and 7.75% respectively. Furthermore, against EKT-based PFL, our approach achieves superior performance with improvements of 0.64%, 2.68%, and 5.82% across the datasets. From an efficiency perspective, Table 2 demonstrates that HEART-PFL requires lower training costs while achieving the highest performance.

4.3. Ablation Study

Component-wise Evaluation in HEART-PFL. To validate the effectiveness of our proposed method, we conduct comprehensive ablation studies on its two key components: HDA and AKT (Table 3). We systematically evaluate the impact of each component on both model performance and the number of trainable parameters. Our approach achieves the highest personalized test accuracy of 63.42% with only 1.46M adapter parameters, demonstrating that HDA and AKT each contribute substantially to the overall performance and that their combination yields optimal results in terms of both accuracy and parameter efficiency.

Out-of-Domain Robustness of HEART-PFL. To evaluate the domain generalization ability of HEART-PFL, we assess its performance when the proxy dataset $\mathcal{D}_\rho$ and the client datasets $\mathcal{D}_k$ originate from different domains. We consider two experimental conditions: (i) distillation using out-of-domain proxy data and (ii) distillation using in-domain proxy data. Under each condition, we further examine two scenarios.

In the first scenario (Figure 3a), the client dataset is CIFAR100 and out-of-domain proxy dataset is Flowers102, our method demonstrates remarkable domain generalization capability. The out-of-domain version achieves 63.01% accuracy while the in-domain version reaches 63.42% accuracy, showing only a marginal 0.41% difference. This minimal gap indicates that HEART-PFL effectively transfers knowledge even across significantly different domains.

In the second scenario (Figure 3b), with Caltech101 as the client dataset and Flowers102 as the out-of-domain proxy dataset, we observe even more compelling results.

Method	CIFAR100			Flowers102			Caltech101		
	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$
FedAvg-per	45.64 ± 0.4	45.44 ± 0.5	45.39 ± 0.2	68.11 ± 1.4	72.68 ± 0.3	72.77 ± 0.6	90.03 ± 0.8	91.89 ± 0.3	92.68 ± 0.7
FedProx-per	45.83 ± 0.9	45.17 ± 0.6	45.50 ± 0.5	68.05 ± 2.1	72.97 ± 0.2	73.50 ± 0.9	89.79 ± 1.0	91.73 ± 0.3	92.63 ± 0.8
FedProto-per	38.89 ± 13.3	41.05 ± 0.5	36.03 ± 0.6	76.87 ± 0.7	63.06 ± 1.2	57.29 ± 0.8	90.77 ± 1.1	86.39 ± 0.9	84.09 ± 0.5
FedPer	56.65 ± 0.3	61.19 ± 0.4	56.74 ± 0.3	81.31 ± 0.8	73.96 ± 0.6	72.08 ± 1.2	93.11 ± 0.7	89.07 ± 0.4	88.03 ± 0.3
LG-FedAvg	43.33 ± 0.3	50.04 ± 0.4	43.69 ± 0.2	76.51 ± 1.1	61.63 ± 1.5	55.41 ± 0.4	91.53 ± 0.1	86.07 ± 0.3	83.57 ± 0.8
Ditto	57.31 ± 9.1	48.72 ± 0.4	45.35 ± 0.2	71.81 ± 1.3	72.92 ± 0.3	73.24 ± 0.7	90.08 ± 1.0	91.92 ± 0.3	92.51 ± 0.8
FedBABU	49.66 ± 2.4	48.40 ± 0.5	45.53 ± 0.2	68.37 ± 1.7	73.30 ± 0.2	73.66 ± 0.6	91.50 ± 0.7	91.90 ± 1.2	92.01 ± 0.7
FedALA	51.44 ± 0.6	54.46 ± 0.5	51.20 ± 0.4	68.63 ± 2.1	72.05 ± 0.4	72.64 ± 1.4	90.86 ± 0.8	92.02 ± 0.7	92.42 ± 0.8
PerAda	62.44 ± 0.4	61.42 ± 0.6	60.40 ± 0.6	80.83 ± 0.6	81.55 ± 0.5	81.55 ± 0.6	90.24 ± 0.8	87.14 ± 4.5	89.68 ± 0.9
HEART-PFL	**63.42** ± 0.1	**61.47** ± 0.2	**61.28** ± 0.4	**84.07** ± 0.5	**83.68** ± 0.1	**84.23** ± 0.2	**95.67** ± 0.3	**94.49** ± 0.3	**94.35** ± 0.2

Table 1. Personalized test accuracy (%) of HEART-PFL and state-of-the-art baselines on CIFAR100, Flowers102, and Caltech101 under Dirichlet client partitions ($\alpha \in \{0.1, 0.3, 0.5\}$). Results are reported as mean±std over three seeds.

Method	Trainable Params
FedAvg-per	11.18M
FedProx-per	11.18M
FedProto-per	11.18M
FedPer	11.18M
LG-FedAvg	11.18M
Ditto	22.36M
FedBABU	11.18M
FedALA	11.18M
PerAda	2.82M
HEART-PFL	**1.46M**

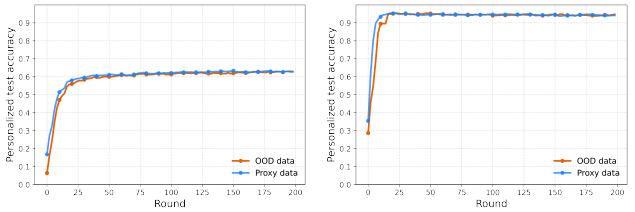
Table 2. Comparison of trainable parameter counts across state-of-the-art PFL baselines under an identical backbone setting. HEART-PFL achieves the lowest parameter cost (1.46M)

Method	Trainable Params	Accuracy (%)
Baseline	11.18M	45.64 ± 0.4
+ HDA	11.18M	58.94 ± 0.5
+ AKT	11.18M	59.46 ± 0.2
+ HDA + AKT	11.18M	61.83 ± 0.5
HEART-PFL (w/ adapter)	**1.46M**	**63.42** ± 0.5

Table 3. Component-wise ablation study for HEART-PFL.

The out-of-domain version achieves 95.40% accuracy, while the in-domain version obtains 95.35% accuracy. In this case, the out-of-domain approach actually outperforms the in-domain method. The HEART-PFL method demonstrates comparable performance regardless of whether the proxy dataset is in-domain or out-of-domain.

Stage-wise HDA Design. Our HDA method aligns personalized prototypes with global features using a dual similarity scheme: cosine similarity in early stage for directional alignment and MSE in deep stage for semantic alignment.



(a) Personalized test accuracy on the CIFAR100 client dataset, using Flowers102 as the out-of-domain proxy dataset.

(b) Personalized test accuracy on the Caltech101 client dataset, using Flowers102 as the out-of-domain proxy dataset.

Figure 3. Out-of-Domain Setting on CIFAR100 and Caltech101. These experiments were conducted using the methods from our HEART-PFL to measure the out-of-domain performance of AKT.

We conduct an ablation study to verify whether this approach ensures optimal alignment at the appropriate stage.

To analyze our method, we employ two metrics. First, we measure Representation Alignment, defined as the cosine similarity between global features and personalized prototypes. Cosine similarity is a primary metric for evaluating semantic alignment [2, 16], which, in our HDA, directly validates its effectiveness in achieving directional consistency. Second, we compute Feature Norm Variance, the standard deviation of feature ℓ_2 norms, measuring how differently the model recognizes and processes various inputs to assess its representation capacity. This is motivated by findings that feature norm distributions encode meaningful model and data characteristics [31].

As shown in Figure 4, personalized test accuracies steadily rise from the 0-layer to the 2-layer configuration, achieving their highest performance with the 2-layer setup. Both alignment and diversity scores show similar trends, indicating that the HDA configuration yields optimal alignment between personalized and global models.

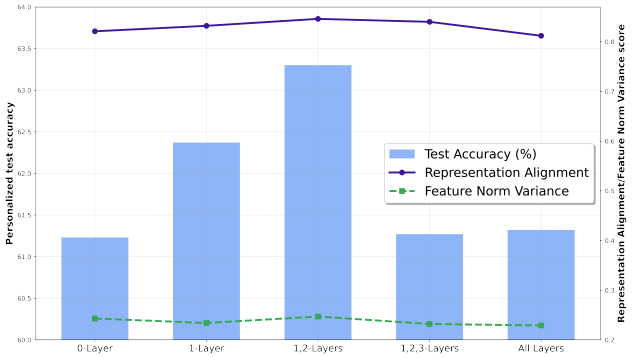
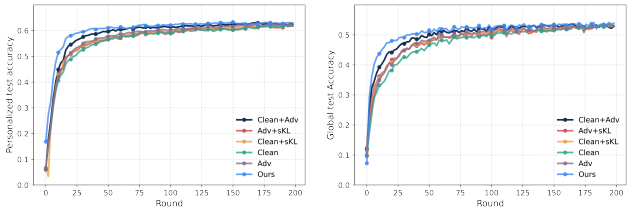


Figure 4. **Layer-wise ablation study of HDA.** We evaluate five configurations: a baseline using MSE loss (without cosine similarity) and settings with cosine similarity applied progressively from one to all layers. The results show that increasing the depth of directional alignment leads to consistent improvements in test accuracy (blue bars), representation alignment (purple line), and feature norm variance (green dashed line).



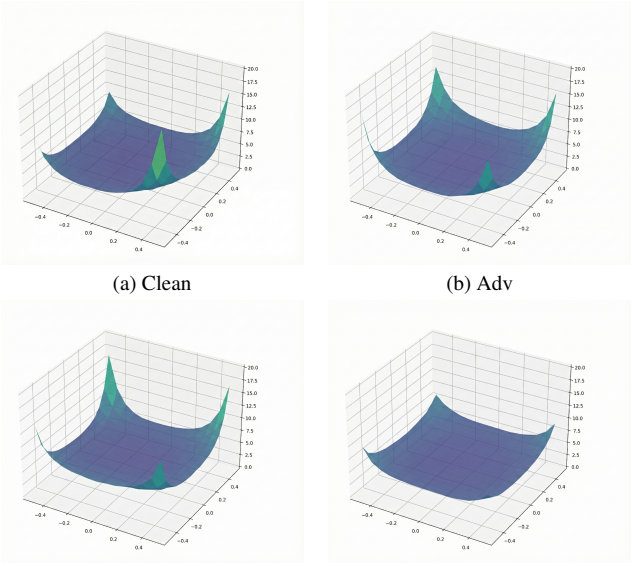
(a) Personalized test performance. (b) Global test performance.

Figure 5. Personalized and global test performance under component ablations of AKT on CIFAR100 with Dirichlet partitions ($\alpha = 0.1$). We compare the AKT (Ours) against variants using only clean samples (Clean), using only adversarial perturbation (Adv), without adversarial perturbation but with symmetric KL (Clean+sKL), without symmetric KL but with adversarial perturbation (Clean+Adv), and without clean samples (Adv+sKL). Across both metrics, the Full AKT configuration achieves the best accuracy.

Component Ablations in AKT. We evaluate how each core component of AKT contributes to performance on CIFAR100 under Dirichlet client partitions with $\alpha = 0.1$. Figure 5a and Figure 5b summarize the personalized and global test accuracies under these ablations.

Results show consistent patterns. First, our AKT outperforms all ablations on both metrics, indicating complementary gains from adversarial perturbation and symmetric KL. In particular, personalized test accuracy reaches 63.42%, the highest among all variants, and global test accuracy reaches 54.08%, also the best. Finally, the gains manifest in both evaluation regimes, indicating that the improvements are not confined to per-client adaptation but also propagate to the aggregated model. Taken together, these results substantiate that AKT is well-suited for PFL, delivering consistent benefits to both personalized and global performance.

To verify that our proxy sample scenario facilitates the optimization stability underlying these performance gains, we examine the 3D loss surfaces of the four ablation variants (Fig. 6). This demonstrates that AKT effectively stabilizes aggregation and reduces bias across heterogeneous clients by reshaping the optimization landscape into a flatter region.



(a) Clean (b) Adv

(c) Clean + Adv (d) AKT (HEART-PFL)

Figure 6. Visualization of the 3D loss landscapes for the four proxy sample scenarios. Each surface is generated by evaluating the average loss around the converged model parameters along two random normalized directions.

5. Conclusion

We propose HEART-PFL to address the limitations of shallow feature alignment and fragile distillation in PFL. On the client side, HDA performs hierarchical alignment by applying cosine similarity to features of the early stage and MSE to the deep stage, preserving client-specific semantics. On the server side, AKT employs symmetric KL distillation on both clean and adversarial proxy samples to stabilize global updates. Experiments on CIFAR100, Flowers102, and Caltech101 show that HEART-PFL surpasses strong baselines with only 1.46M trainable parameters while maintaining robustness under out-of-domain proxy data. Ablation studies further demonstrate that HDA and AKT provide complementary benefits in alignment, robustness, and optimization stability. We conclude that HEART-PFL establishes a principled foundation for robust PFL under heterogeneity.

References

[1] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Ankit Kumar Singh, and Sunav Choudhury. Federated learning with personalization layers. 2019. 1, 3, 6

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1608. PMLR, 2020. 2, 7

[3] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International conference on machine learning*, pages 2089–2099. PMLR, 2021. 1

[4] Jian-hui Duan, Wenzhong Li, Derun Zou, Ruichen Li, and Sanglu Lu. Federated learning with data-agnostic distribution fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8074–8083, 2023. 1

[5] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer vision and Image understanding*, 106(1):59–70, 2007. 6

[6] Liang Gao, Huazhu Fu, Li Li, Yingwen Chen, Ming Xu, and Cheng-Zhong Xu. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10112–10121, 2022. 1, 6

[7] Fucheng Guo, Zeyu Luan, Qing Li, Dan Zhao, and Yong Jiang. Fedorgp: Guiding heterogeneous federated learning with orthogonality regularization on global prototypes. *arXiv preprint arXiv:2502.16119*, 2025. 1, 2, 3

[8] Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020. 1

[9] Filip Hanzely, Slavomír Hanzely, Samuel Horváth, and Peter Richtárik. Lower bounds and optimal algorithms for personalized federated learning. *Advances in Neural Information Processing Systems*, 33:2304–2315, 2020. 1

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 5

[11] Hai Jin, Dongshan Bai, Dezhong Yao, Yutong Dai, Lin Gu, Chen Yu, and Lichao Sun. Personalized edge intelligence via federated self-knowledge distillation. *IEEE Transactions on Parallel and Distributed Systems*, 34(2):567–580, 2022. 1

[12] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International conference on machine learning*, pages 5132–5143. PMLR, 2020. 2

[13] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009. 6

[14] Hyoje Lee, Yeachan Park, Hyun Seo, and Myungjoo Kang. Self-knowledge distillation via dropout. *Computer Vision and Image Understanding*, 233:103720, 2023. 5

[15] Dali Li and Junpu Wang. Fedmd: Heterogeneous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019. 2

[16] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10713–10722, 2021. 2, 7

[17] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020. 6

[18] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine learning and systems*, pages 429–450, 2020. 2

[19] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International conference on machine learning*, pages 6357–6368. PMLR, 2021. 1, 3, 6

[20] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019. 1

[21] Xin-Chun Li, De-Chuan Zhan, Yunfeng Shao, Bingshuai Li, and Shaoming Song. Fedphp: Federated personalization with inherited private models. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 587–602. Springer, 2021. 1, 3

[22] Paul Pu Liang, Terrance Liu, Liu Ziyin, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 2020. 3, 6

[23] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in neural information processing systems*, 33:2351–2363, 2020. 2, 5

[24] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems*, pages 2351–2363, 2020. 2

[25] Shaoming Liu, Yunfeng Zhang, et al. Global prototype distillation for heterogeneous federated learning. *Scientific Reports*, 14:12284, 2024. 1, 2, 3

[26] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 5

[27] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020. 1

[28] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 1, 2, 6

[29] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In

Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP), 2008. 6

[30] Jaehoon Oh, Sangmook Kim, and Se-Young Yun. Fedbabu: Towards enhanced representation for federated image classification. *arXiv preprint arXiv:2106.06042*, 2021. 1, 6

[31] Jaewoo Park, Sae-Young Chun, and Nojun Kwak. Understanding the feature norm for out-of-distribution detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 21544–21554, 2023. 7

[32] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022. 1, 2, 6

[33] Yafei Qi, Chen Wang, Zhaoning Zhang, Yaping Liu, and Yongmin Zhang. Balance divergence for knowledge distillation. *arXiv preprint arXiv:2501.07804*, 2025. 3, 5

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 6

[35] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in neural information processing systems*, 33:21394–21405, 2020. 1

[36] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. In *Advances in Neural Information Processing Systems*, pages 21394–21405, 2020. 3

[37] Rishub Tamirisa, Chulin Xie, Wenxuan Bao, Andy Zhou, Ron Arel, and Aviv Shamsian. Fedselect: Personalized federated learning with customized selection of parameters for fine-tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23985–23994, 2024. 3, 6

[38] Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. Fedproto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8432–8440, 2022. 1, 2, 3, 6

[39] Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. Rethinking kullback-leibler divergence in knowledge distillation for large language models. *arXiv preprint arXiv:2404.02657*, 2024. 3, 5

[40] Chulin Xie, De-An Huang, Wenda Chu, Daguang Xu, Chaowei Xiao, Bo Li, and Anima Anandkumar. Perada: Parameter-efficient federated learning personalization with generalization guarantees. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 23838–23848, 2024. 1, 2, 3, 5, 6

[41] Yuanhao Xiong, Ruochen Wang, Minhao Cheng, Felix Yu, and Cho-Jui Hsieh. Feddm: Iterative distribution matching for communication-efficient federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16323–16332, 2023. 1

[42] Dawei Yao, Chang Zhou, Botong Chew, and Xinyu Liang. Fedgkd: Global knowledge distillation in federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(4):4717–4729, 2024. 2

[43] Jie Zhang, Song Guo, Xiaosong Ma, Haozhao Wang, Wenchao Xu, and Feijie Wu. Parameterized knowledge transfer for personalized federated learning. *Advances in Neural Information Processing Systems*, 34:10092–10104, 2021. 1

[44] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, Jian Cao, and Haibing Guan. Gpfl: Simultaneously learning global and personalized feature information for personalized federated learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5041–5051, 2023. 1

[45] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedala: Adaptive local aggregation for personalized federated learning. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11237–11244, 2023. 3, 6

[46] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedcp: Separating feature information for personalized federated learning via conditional policy. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pages 3249–3261, 2023. 1