Federated Representation Angle Learning*

Liping Yi[1], Han Yu[2], Gang Wang[1,*], Xiaoguang Liu[1,*], Xiaoxiao Li[3,4]
[1]*College of Computer Science, TMCC, SysNet, DISSec, GTIISC, Nankai University, Tianjin, China*
[2]*College of Computing and Data Science, Nanyang Technological University, Singapore*
[3]*Department of Electrical and Computer Engineering, University of British Columbia, Vancouver, Canada*
[4]*Vector Institute, Vancouver, Canada*
{yiliping, wgzwp, liuxg}@nbjl.nankai.edu.cn, han.yu@ntu.edu.sg, xiaoxiao.li@ece.ubc.ca

Abstract

*Model-heterogeneous federated learning (MHFL) is a challenging FL paradigm designed to allow FL clients to train structurally heterogeneous models under the coordination of an FL server. Existing MHFL methods face significant limitations when it comes to transferring global knowledge to clients as a result of sharing only partial homogeneous model parameters or calculating distance loss, leading to inferior model generalization. To bridge this gap, we propose a novel model-heterogeneous Federated learning method with Representation Angle Learning (*FedRAL*). It consists of three innovative designs: (1) We first introduce representation angle learning into MHFL. Specifically, we embed a homogeneous square matrix into the local heterogeneous model of each client, which learns the angle information of local representations. These homogeneous representation angle square matrices are aggregated on the server to fuse representation angle knowledge shared by clients for enhancing the generalization of local representations. (2) As different clients might have heterogeneous system resources, we propose an adaptive diagonal sparsification strategy to reduce the numbers of the parameters of representation angle square matrices uploaded to the server, to improve FL communication efficiency. (3) To enable effective fusion of sparsified homogeneous local representation angle square matrices, we design an element-wise weighted aggregation approach. Experiments on* 4 *benchmark datasets under* 2 *types of non-IID divisions over* 6 *state-of-the-art baselines demonstrate that* FedRAL *achieves the best performance. It improves test accuracy, communication efficiency and computational efficiency by up to* 5.03%, $12.43\times$ *and* $6.49\times$, *respectively.*

*Corresponding author.

1. Introduction

Federated learning (FL) [13, 14, 24, 36, 45, 55, 56] is a privacy-preserved distributed machine learning paradigm. Typically, an FL server is often involved to broadcast a global model to FL clients. The clients then further train the global model on local data to obtain local models. The server aggregates local model updates received from the clients to produce a new global model. In this way, client data are not exposed. Under this setting, the models trained by the clients must follow the same structure.

In practice, clients participating in FL often have non-independent and identically distributed (non-IID) data, *a.k.a.*, data heterogeneity [27, 42, 50, 58, 75, 78, 79], the local models trained on such data are biased, so the global model obtained by aggregating them may not perform well on all client data. Besides, FL clients also have heterogeneous system configurations in terms of computing power and communication bandwidth, *a.k.a.*, system heterogeneity [39, 57, 59, 60, 67]. Thus, requesting all clients to train the same model might not be viable for some weaker clients, while leaving stronger clients under-utilized. More critically, some companies or institutions joining FL as clients might face intellectual property issues regarding their proprietary local models. Thus, they might be reluctant to share their private heterogeneous models, *a.k.a.*, model heterogeneity. These challenges have inspired the field of model-heterogeneous federated learning (MHFL) with preserved data and model privacy, which advocates avoiding directly sharing private local heterogeneous models [61–65].

Existing MHFL approaches mainly include: a) Model-decoupling: each client model is decoupled into heterogeneous and homogeneous parts, with only homogeneous parts being shared with the server [11, 30]. b) Knowledge distillation: the server aggregates local representation or logits from different clients to generate global representation or logits, which is used to calculate distillation loss for local training [28, 31]. c) Mutual training: each client is

assigned a shared homogeneous small model and trains it with the local heterogeneous model by mutual loss between them [49, 53]. Although these methods support model heterogeneity, their capability for transferring global knowledge from the server to clients is limited due to sharing partial model parameters or using distillation loss, thereby constraining the generalization of local models. Whereas, model representations often involve more sufficient semantic knowledge than partial model parameters or distillation loss, which are explored in this work for knowledge sharing.

A recent work [44] empirically verified that parameter fine-tuning approaches (*e.g.*, LoRA [18]) cannot sufficiently retain semantic information of the pre-trained model. Whereas, fine-tuning the angles (*i.e.*, gradient directions) of the pre-trained model can achieve this, thereby enhancing the generalization of the pre-trained model. The work further proposed the Orthogonal Fine-Tuning (OFT) method to fine-tune the parameter angles of the pre-trained models via an orthogonal matrix (Figure 1). In short, parameter angles involve sufficient semantic information that can enhance model generalization. Therefore, we can enhance FL model generalization through sharing parameter angles instead of model parameters. However, the FL server can not aggregate parameter angles from structure-heterogeneous client models. Considering representations also with sufficient semantic knowledge, we attempt to share homogeneous representation angles among clients to achieve generalization-enhanced model-heterogeneous FL.

To this end, we propose a novel model-heterogeneous Federated learning method with Representation Angel Learning (`FedRAL`) to enhance the transfer of generalizable global knowledge from the FL server to FL clients. It consists of three innovative designs: (1) Since representations are extracted high-level features from data, with sufficient semantic information, `FedRAL` embeds a homogeneous square matrix into the local heterogeneous model to learn the angle information of local extracted representations. The server aggregates local homogeneous representation angle square matrices from different clients to generate a global representation angle square matrix, thereby fusing cross-client representation angle knowledge. Then, the global representation angle square matrix with sufficient generalized representation angle information is used to enhance the generalization of local representations during local model training. Different from the strong assumption of the orthogonality of the matrix in OFT, the homogeneous representation angle square matrix is initialized randomly. (2) Considering the diverse system resources (*e.g.*, communication bandwidth) of different clients, we design an adaptive diagonal sparsification strategy for the homogeneous representation angle square matrix that allows clients to upload diagonal blocks with sizes matching with communication capacity. (3) To effectively aggregate sparsified local homogeneous representation angle square matrices on the server, element-wise weighted aggregation is designed.

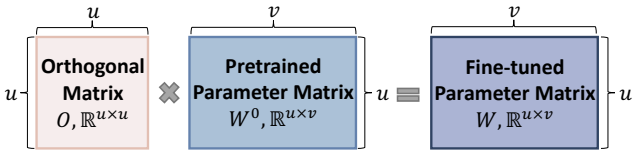

Figure 1. Orthogonal Fine-Tuning (OFT).

Experiments on 4 benchmark datasets under 2 modes of non-IID divisions against 6 advanced MHFL methods demonstrate the superiority of `FedRAL`. Compared with the state-of-the-art baseline, `FedRAL` improves test accuracy by up to 5.03%, while enhancing communication and computational efficiency by up to $12.43\times$ and $6.49\times$.

2. Related Work

Based on different approaches used for achieving MHFL, existing methods can be divided into 4 categories.

Heterogeneous Subnet. Some works [3–5, 12, 17, 33, 66, 77] assume that each client trains local heterogeneous subnets of the global model, and the server aggregates them according to parameter positions to re-construct a new global model. They allow each client to train personalized heterogeneous subnets to tackle data and system heterogeneity. However, uploading subnets exposes model parameters, thereby breaching model IP protection requirements.

Model Decoupling. These methods [7, 11, 22, 30, 32, 38, 41, 62] share partial local model parameters instead of the entire local model for global aggregation. Specifically, they decouple each client's local heterogeneous model into heterogeneous and homogeneous parts, and only share homogeneous parts for cross-client knowledge exchange. Nevertheless, sharing partial model parameters still violates model IP protection needs. Furthermore, the unshared remaining parameters might overfit due to local training.

Knowledge Distillation. These methods [1, 2, 6, 8, 9, 15, 16, 19–21, 23, 28, 29, 31, 34, 37, 40, 46, 47, 51, 52, 68, 69, 71–73, 76] aggregate the local seen-class representations or logits extracted on local private data or a public dataset to generate the global representation or logits by class, which is used to calculate distillation loss with the local representations or logits. Although they protect local model structures, the distillation loss only transfers limited knowledge from the server to clients. Besides, sharing class information might be prohibited in some data privacy-sensitive applications. The public dataset used must follow similar data distributions with local private data, which is difficult to access due to data privacy.

Mutual Learning. These methods [25, 35, 43, 49, 53] add a global homogeneous small model shared by all clients, which is alternately trained with the local heterogeneous model by the mutual loss between the outputs of

the two models for each sample. Although they hide local heterogeneous model structures, the mutual loss only transfers limited knowledge between the two models. The unstable mutual loss in early training rounds might result in both models not converging.

Existing MHFL methods generally transfer limited generalized knowledge from the server to clients, leading to model performance bottlenecks. FedRAL effectively tackles data, system and model heterogeneity simultaneously, allowing clients to share a global homogeneous representation angle square matrix for learning generalized representation angle information. It bridges this important gap by enhancing the generalization of local representations via utilizing the global homogeneous representation angle square matrix to fine-tune local representation angles, while preserving personalized semantic information.

3. Preliminaries

3.1. Orthogonal Fine-Tuning

Fine-tuning the parameters of a pre-trained model has been demonstrated to compromise its semantic information, while fine-tuning its parameter angles (*i.e.*, gradient directions) can sufficiently retain its original semantic information [44]. This enables the preservation of the generalized ability when adapting to downstream tasks. Hypersphere Energy (HE) measures the uniformity of neuron distribution on the unit hypersphere. A lower HE value is beneficial to keeping pre-trained model generalization ability. To minimize HE variance between the pre-trained model $W^0 \in \mathbb{R}^{u\times v}$ and the angle-fine-tuned model $W \in \mathbb{R}^{u\times v}$, orthogonal finetuning (OFT) [44] fine-tunes the parameter angles of the pre-trained model through an orthogonal matrix $O \in \mathbb{R}^{u\times u}$, $W = O \times W^0$, as depicted in Figure 1. To ensure orthogonality, OFT utilizes Cayley parameterization to construct matrix $O = (I + Q)(I - Q)^{-1}$, where $Q \in \mathbb{R}^{u\times u}$ is a skew-symmetric matrix satisfying $Q = -Q^T$. Due to inverse operations, updating the orthogonal matrix O during fine-tuning introduces high computational overhead, especially with a large dimension u.

Unlike OFT, we explore fine-tuning the angles of representations with sufficient semantic information. We relax the orthogonality assumption and avoid expensive inverse operations. Instead, we randomly initialize the homogeneous representation angle square matrix, which is embedded into the local heterogeneous model. They are trained in an end-to-end manner to improve computation efficiency.

3.2. Problem Formulation

In FedRAL, a server coordinates K clients with heterogeneous local models for collaborative training. Client k's local model $f_k(\omega_k)$ consists of two parts: 1) a heterogeneous feature extractor $f_k^{ex}(\omega_k^{ex})$, and 2) a homogeneous prediction head $f^{hd}(\omega_k^{hd})$ (all clients conduct the same prediction task), $f_k(\omega_k) = f_k^{ex}(\omega_k^{ex}) \circ f^{hd}(\omega_k^{hd})$, as depicted in Figure 2. Each client trains a local homogeneous square matrix $\boldsymbol{\mathcal{A}}_k$ to learn the angle information of local representations extracted by the heterogeneous feature extractor. The server aggregates local homogeneous representation angle square matrices from different clients to generate a global representation angle square matrix $\boldsymbol{\mathcal{A}}$ for cross-client representation angle knowledge fusion. Therefore, the training objective of FedRAL is to minimize the loss sum of all heterogeneous client models with the help of the global homogeneous representation angle square matrix $\boldsymbol{\mathcal{A}}$:

$$\min_{\boldsymbol{\mathcal{A}},\{\omega_0,\ldots,\omega_{K-1}\}} \sum_{k=0}^{K-1} \ell\left(\mathcal{F}_k\left(D_k; (\omega_k \circ \boldsymbol{\mathcal{A}})\right)\right), \qquad (1)$$

where ℓ is loss. $\mathcal{F}_k(\omega_k \circ \boldsymbol{\mathcal{A}})$ denotes the combination of the local heterogeneous model ω_k and the global representation angle square matrix $\boldsymbol{\mathcal{A}}$, D_k is the non-IID data of client k.

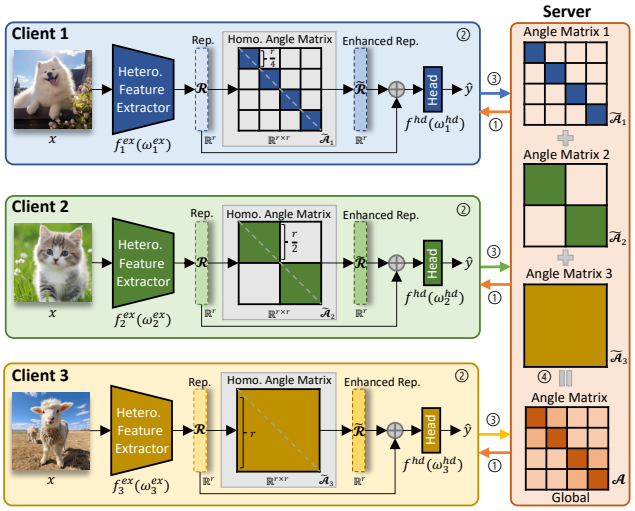

Figure 2. The workflow of FedRAL.

4. The Proposed FedRAL Approach

FedRAL consists of three innovative designs: (1) homogeneous representation angle square matrix learning, (2) adaptive diagonal sparsification, and (3) element-wise weighted homogeneous square matrix aggregation. The workflow of FedRAL in a single communication round includes the following steps.

Overview. As shown in Figure 2, in the t-th communication round, FedRAL executes 4 steps:

① The server broadcasts the latest global representation angle square matrix $\boldsymbol{\mathcal{A}}^{t-1}$ to K participating clients.

② A client k embeds the global representation angle square matrix $\boldsymbol{\mathcal{A}}^{t-1}$ into its local model $f_k(\omega_k^{t-1})$ to enhance

the generalization of local representations, while learning the personalized angle information of local representations via end-to-end training.

③ The local homogeneous representation angle square matrix $\mathcal{A}_k^t$ after local training is sparsified into $\tilde{\mathcal{A}}_k^t$ via the proposed adaptive diagonal sparsification strategy before being uploaded to the server.

④ The server aggregates sparsified local homogeneous representation angle square matrices $\{\tilde{\mathcal{A}}_0^t, \ldots, \tilde{\mathcal{A}}_{K-1}^t\}$ via the proposed element-wise weighted aggregation to generate a new global representation angle square matrix $\mathcal{A}^t$.

Inference. These steps are iteratively executed until heterogeneous client models $\{f_0(\omega_0), \ldots, f_{K-1}(\omega_{K-1})\}$ converge, which are used for inference. Algorithm 1 describes the detailed workflow of `FedRAL`.

4.1. Representation Angle Learning

During the t-th training round, a client k uses its local heterogeneous feature extractor $f_k^{ex}(\omega_k^{ex,t-1})$ to extract the personalized representation $\mathcal{R} \in \mathbb{R}^r$ of the input sample $\boldsymbol{x}$:

$$\mathcal{R} = f_k^{ex}(\boldsymbol{x}; \omega_k^{ex,t-1}). \tag{2}$$

Then, the extracted personalized representation $\mathcal{R} \in \mathbb{R}^r$ is multiplied by the received global representation angle square matrix $\mathcal{A}^{t-1} \in \mathbb{R}^{r\times r}$ to produce the representation $\tilde{\mathcal{R}} \in \mathbb{R}^r$ with enhanced generalization:

$$\tilde{\mathcal{R}} = \mathcal{R} \times \mathcal{A}^{t-1}. \tag{3}$$

Then, $\mathcal{R} \in \mathbb{R}^r$ and $\tilde{\mathcal{R}} \in \mathbb{R}^r$ are summed and fed into the prediction head to make a prediction:

$$\hat{y} = f^{hd}(\mathcal{R} + \tilde{\mathcal{R}}; \omega_k^{hd,t-1}). \tag{4}$$

The loss ℓ (*e.g.*, Cross-Entropy loss [74]) between the prediction $\hat{y}$ and the ground-truth label y is used to update the heterogeneous client model $f_k(\omega_k^{t-1})$ and the homogeneous representation angle square matrix $\mathcal{A}^{t-1}$ simultaneously via end-to-end training:

$$\begin{aligned} f_k(\omega_k^t) &\leftarrow f_k(\omega_k^{t-1}) - \eta_\omega \nabla \ell(\hat{y}, y), \\ \mathcal{A}_k^t &\leftarrow \mathcal{A}^{t-1} - \eta_{\mathcal{A}} \nabla \ell(\hat{y}, y), \end{aligned} \tag{5}$$

where η_ω and $\eta_{\mathcal{A}}$ are the learning rates of the local heterogeneous model and the homogeneous representation angle square matrix. We set $\eta_\omega = \eta_{\mathcal{A}}$ by default as they are updated at the same time.

The shared global homogeneous representation angle square matrix serves three functions during local training: a) it transfers globally generalizable representation angle knowledge from the server to the client by producing the generalization-enhanced representation; b) it learns the personalized representation angle knowledge from the extracted personalized representations, thereby transferring local personalized representation angle knowledge to the server; and c) the summed representation includes both generalized and personalized knowledge, thereby improving model expressiveness while addressing non-IID problem.

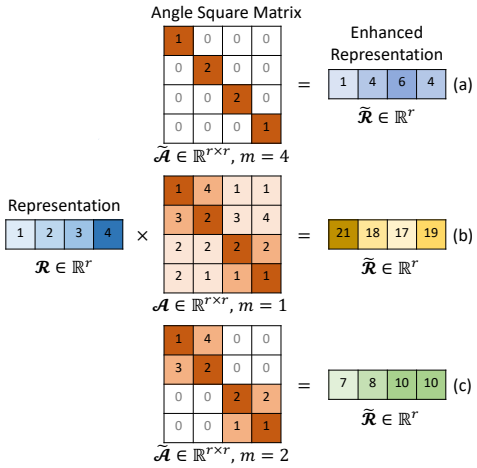

Figure 3. An illustrative example of diagonal sparsification.

4.2. Adaptive Diagonal Sparsification

To improve communication efficiency while maintaining model performance, we design an adaptive diagonal sparsification strategy to sparsify the local representation angle square matrices before uploading them to the FL server.

Each element in $\mathcal{R} \in \mathbb{R}^r$ corresponds to the element on the diagonal of $\mathcal{A} \in \mathbb{R}^{r\times r}$. If we only retain the diagonal elements of $\mathcal{A}$ and set other elements to be 0, the angle of each element in the representation is directly adjusted by the corresponding element in the square matrix diagonal, as depicted in Figure 3(a). When all elements of the square matrix are non-zero, non-diagonal elements might disturb representation angle adjustment due to multiplication with non-corresponding representation elements. As Figure 3(b) shows, the enhanced representation processed by the entire representation angle square matrix is significantly different from the one processed by the sparse square matrix with only diagonal elements.

Since adjacent elements in the representation are semantically correlated, retaining partial elements wrapped with the diagonal in the representation angle square matrix can enhance the angle correlation of adjacent representation elements. Therefore, the proposed sparsification strategy retains the elements from the diagonal blocks of the square matrix and masks other elements with 0, as shown in Figure 3(c). For a $\mathbb{R}^{r\times r}$ square matrix, m diagonal blocks are retained. Each diagonal block is $\mathbb{R}^{\frac{r}{m}\times\frac{r}{m}}$.

Clients with different communication bandwidths can select different numbers of diagonal blocks m_k for the local homogeneous representation angle matrix $\mathcal{A}_k^t$ to produce

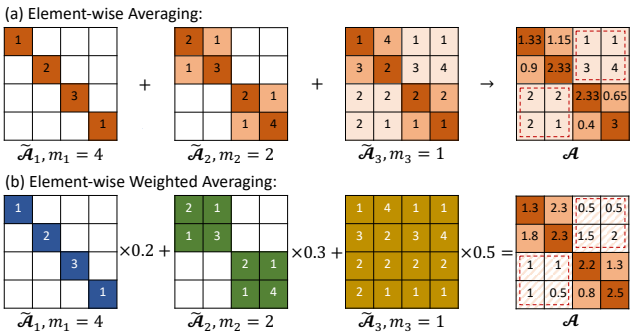

Figure 4. Element-wise averaging (a) without or (b) with weights.

the sparsified representation angle square matrix $\tilde{\mathcal{A}}_k^t$:

$$\tilde{\mathcal{A}}_k^t = \text{Diagonal-Sparsification}(\mathcal{A}_k^t; m_k). \tag{6}$$

4.3. Element-wise Weighted Aggregation

After receiving the sparsified local homogeneous representation square matrices from K clients in the t-th communication round, the server aggregates them according to the element positions with weighted averaging:

$$\mathcal{A}^t = \sum_{k=0}^{K-1} \frac{n_k}{n} \tilde{\mathcal{A}}_k^t, \tag{7}$$

where n_k is the data volume of the client k, and n is the total data volume of K clients.

This element-wise weighted aggregation rule provides two benefits. a) All clients upload the diagonal elements of local homogeneous representation angle square matrices. Using $\frac{n_k}{n}$ to weigh the diagonal elements can sufficiently reflect the overall data distribution across all clients, improving the generalization of the newly generated global square matrix. b) As the example given in Figure 4(a), for non-diagonal elements, if we directly average overlapped them by element, when one non-diagonal element is only provided by one client, then it is the value of this non-diagonal element. In this case, these non-overlapped non-diagonal elements may negatively affect the clients who do not provide for these elements. Instead, as shown in Figure 4(b), weighted averaging scales down such elements and then reduce negative effects while improving the generalization of the newly generated global square matrix.

5. Experimental Evaluation

To evaluate the performance of `FedRAL` [1], we compare it with 6 state-of-the-art existing approaches on 4 widely adopted benchmark datasets under 2 types of data heterogeneity settings. We implement `FedRAL` and baselines using pytorch and run them on 4 NVIDIA GeForce RTX 3090 GPUs (24GB memory).

[1] https://github.com/LipingYi/FedRAL

Algorithm 1: `FedRAL`

Input: K, total number of participating clients;
T, total number of communication rounds;
η_ω, learning rate of heterogeneous client models;
$\eta_{\mathcal{A}}$, learning rate of the homogeneous representation angle square matrix.
Randomly initialize the global representation angle square matrix $\mathcal{A}^0$ and heterogeneous client models $[f_0(\omega_0^0), \ldots, f_k(\omega_k^0), \ldots, f_{K-1}(\omega_{K-1}^0)]$.
for $t = 1$ ***to*** T **do**
  /* **FL Server**: */
  Broadcast $\mathcal{A}^{t-1}$ to participating clients;
  /* **Each FL Client** k: */
  // Representation Angle Matrix Learning
  for *each* $(\boldsymbol{x}_i, y_i) \in D_k$ **do**
    $\mathcal{R}_i = f_k^{ex}(\boldsymbol{x}_i; \omega_k^{ex,t-1})$;
    $\tilde{\mathcal{R}}_i = \mathcal{R}_i \times \mathcal{A}^{t-1}$;
    $\hat{y}_i = f^{hd}(\mathcal{R}_i + \tilde{\mathcal{R}}_i; \omega_k^{hd,t-1})$;
    $f_k(\omega_k^t) \leftarrow f_k(\omega_k^{t-1}) - \eta_\omega \nabla \ell(\hat{y}_i, y_i)$;
    $\mathcal{A}_k^t \leftarrow \mathcal{A}^{t-1} - \eta_{\mathcal{A}} \nabla \ell(\hat{y}_i, y_i)$;
  end
  // Adaptive Diagonal Sparsification
  Obtain $\tilde{\mathcal{A}}_k^t$ using Eq. (6);
  Upload $\tilde{\mathcal{A}}_k^t$ to the FL server;
  /* **FL Server**: */
  // Element-wise Weighted Aggregation
  Generate $\mathcal{A}^t$ by aggregation using Eq. (7);
end
Return local heterogeneous client models $\{f_0(\omega_0^{T-1}), \ldots, f_k(\omega_k^{T-1}), \ldots, f_{K-1}(\omega_{K-1}^{T-1})\}$.

5.1. Experiment Setup

We first introduce datasets, base models, comparison baselines, evaluation metrics and hyperparameter settings.

Datasets. We choose 4 image classification benchmark datasets commonly used in FL investigations.

- **Fashion-MNIST**[2] [54] contains $60,000$ training and $10,000$ testing grayscale 28×28 10-class images.
- **CIFAR-10**[3] [26] contains $50,000$ training and $10,000$ testing colourful 32×32 10-class images.
- **CIFAR-100**[4] [26] contains $50,000$ training and $10,000$ testing colourful 32×32 100-class images.
- **Tiny-ImageNet**[5] [10] contains $100,000$ training and $10,000$ testing colourful 64×64 200-class images.

[2] https://github.com/zalandoresearch/fashion-mnist
[3] https://www.cs.toronto.edu/%7Ekriz/cifar.html
[4] https://www.cs.toronto.edu/%7Ekriz/cifar.html
[5] https://tiny-imagenet.herokuapp.com/

Table 1. Heterogeneous CNNs for Fashion-MNIST.

Layer	CNN-1	CNN-2	CNN-3	CNN-4	CNN-5
Conv1	5×5, 20	5×5, 20	5×5, 20	5×5, 20	5×5, 20
Maxpool1	2×2	2×2	2×2	2×2	2×2
Conv2	5×5, 20	5×5, 20	5×5, 20	5×5, 20	5×5, 20
Maxpool2	2×2	2×2	2×2	2×2	2×2
FC1	300	200	150	100	50
FC2	50	50	50	50	50
FC3	10	10	10	10	10
model size	0.47 MB	0.33 MB	0.26 MB	0.19 MB	0.12 MB

Note: kernel size: 5×5, with 20 filters for convolution layers.

Table 2. Heterogeneous CNNs for CIFAR-10 and CIFAR-100.

Layer	CNN-1	CNN-2	CNN-3	CNN-4	CNN-5
Conv1	5×5, 16	5×5, 16	5×5, 16	5×5, 16	5×5, 16
Maxpool1	2×2	2×2	2×2	2×2	2×2
Conv2	5×5, 32	5×5, 16	5×5, 32	5×5, 32	5×5, 32
Maxpool2	2×2	2×2	2×2	2×2	2×2
FC1	2000	2000	1000	800	500
FC2	500	500	500	500	500
FC3	10/100	10/100	10/100	10/100	10/100
model size	10.01 MB	6.93 MB	5.04 MB	4.05 MB	2.56 MB

Note: kernel size: 5×5, with 16 or 32 filters for convolution layers.

Then, we partition them into non-IID modes in 2 ways:

- **Pathological.** Following [48], we allocate $\{2, 2, 10, 20\}$ classes from $\{10, 10, 100, 200\}$ classes to each FL client.
- **Practical.** Following [43], we allocate $\{10, 10, 100, 200\}$ classes to each FL client and use a Dirichlet distribution function with a hyperparameter $\alpha = 0.4$ to generate diverse ratios of one class for different clients.

Base Models. Following [62], we construct 5 heterogeneous CNN models {CNN-1, ..., CNN-5} with different channels of convolutional layers or dimensions of linear layers (Tables 1 and 2) and allocate them to 100 clients. The client k is assigned CNN-($k\%5$). Note that the representation $\mathcal{R}$ dimension r (the dimension of the penultimate linear layer) is $\{50, 500, 500\}$ for Fashion-MNIST, CIFAR-10 and CIFAR-100. Thus, the dimensions of the homogeneous representation angle square matrix $\mathcal{A}$ are $\{50 \times 50, 500 \times 500, 500 \times 500\}$. For Tiny-ImageNet, we evenly allocate ResNet-$\{18, 34, 50, 101, 152\}$ to 100 clients.

Comparison Baselines. We compare `FedRAL` with 6 baselines, which are the latest representative MHFL works that support completely heterogeneous local models.

- `Standalone`: each client solely trains its local heterogeneous model, without FL.
- **Model Decoupling:** `LG-FedAvg` [30].
- **Mutual Learning:** `FedKD` [53], `FedAPEN` [43].
- **Knowledge Distillation without Public Dataset:** `FedProto` [51], `FedTGP` [72].

Evaluation Metrics. We evaluate the performances of `FedRAL` and baselines with the following three metrics:

- **Model Accuracy.** We record each client's model accuracy (%) and calculate their average in each communication round. We report the highest average accuracy.
- **Communication Cost.** We track the average number of transmitted parameters between clients and the server in one communication round, and we record the total rounds consumed to reach the specified target accuracy. The product of the two is the total communication cost.
- **Computational Overhead.** We track the average computational FLOPs of clients in one communication round, and we record the total communication rounds consumed to reach the specified target accuracy. The product of the two is the total computational overhead.

Hyperparameters. We use grid-search to choose the optimal hyperparameters for all algorithms. For general FL hyperparameters, we set the local training epoch E as $\{1, 10\}$ and batch size B as $\{32, 64, 128, 256, 512\}$, and we choose the SGD gradient optimizer with a learning rate of 0.01. To ensure the sufficient convergence of algorithms, we set the total communication rounds T as 500. For the unique hyperparameter - the number m of diagonal blocks for sparsifying the homogeneous representation angle square matrix in `FedRAL`, we set it to be the factor of the representation dimension r, *e.g.*, $m = \{1, 2, \ldots, 50\}$ for Fashion-MNIST, and $m = \{1, 2, \ldots, 500\}$ for CIFAR.

5.2. Comparison Results Analysis

Average Accuracy. Table 3 shows that `FedRAL` outperforms all baselines in terms of average test accuracy, demonstrating its effectiveness in enhancing local model generalization through the shared homogeneous representation angle square matrix. As a recent MHFL method, `FedTGP` achieves the second-best performance. We notice that `FedAPEN` fails to converge in some cases. This can be attributed to the fact that it uses the same coefficients to combine the outputs of the heterogeneous and homogeneous models for all data samples, failing to balance personalization and generalization under varying data features.

Figure 5 presents how the average test accuracies of `FedRAL` and the state-of-the-art `FedTGP` baseline vary as communication rounds. We see that `FedRAL` converges to a higher accuracy with a faster speed than `FedTGP`, especially under more non-IID practical FL settings. This demonstrates the effectiveness of `FedRAL` in alleviating data heterogeneity while supporting the collaboration of heterogeneous client models.

Communication Cost. Table 4 presents the communication costs of the state-of-the-art `FedTGP` baseline and `FedRAL`. We can observe that `FedRAL` consumes a lower single-round communication cost than `FedTGP` (column-3), since the sparse representation angle square matrix communicated in `FedRAL` has fewer parameters than the label-wise average representations transmitted in `FedTGP`. `FedRAL` also requires fewer communication rounds to reach the specified target accuracy (column-5), since the de-

Table 3. Comparison of average test accuracy (%) on 4 datasets under pathological and practical data heterogeneity settings.

Method	Pathological				Practical			
	Fashion-MNIST	CIFAR-10	CIFAR-100	Tiny-ImageNet	Fashion-MNIST	CIFAR-10	CIFAR-100	Tiny-ImageNet
Standalone	99.03±0.53	91.97±0.37	53.04±0.24	33.15±0.02	73.39±0.89	40.72±0.83	9.56±0.34	7.35±0.21
LG-FedAvg [30]	98.87±0.40	91.27±0.25	45.83±0.67	35.79±0.19	71.97±0.25	40.44±0.57	9.80±0.39	7.51±0.34
FedKD [53]	58.37±0.05	73.21±0.56	37.21±0.98	34.53±0.16	45.57±0.49	28.14±0.29	8.29±0.28	6.29±0.23
FedAPEN [43]	44.20±0.58	-	-	-	-	-	-	-
FedProto [51]	99.02±0.50	92.49±0.26	53.67±0.79	36.24±0.27	74.06±0.63	40.20±0.11	9.27±0.71	7.63±0.29
FedTGP [72]	99.09±0.65	92.50±0.31	53.20±0.78	36.92±0.10	73.64±0.60	40.95±0.09	9.37±0.13	7.65±0.19
FedRAL	**99.54**±0.16	**93.60**±0.19	**58.70**±0.05	**38.75**±0.13	**78.58**±0.12	**42.97**±0.10	**11.70**±0.02	**9.47**±0.13

Note: "-" denotes failure to converge. Bold indicates the highest accuracy, and underline indicates the second highest accuracy.

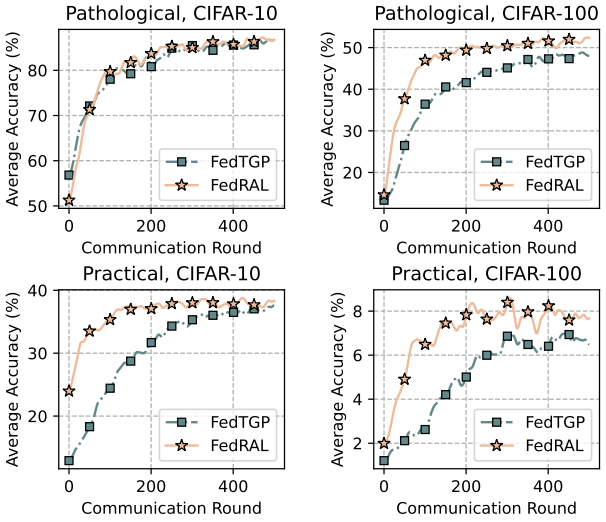

Figure 5. Average test accuracy (%) vs. number of rounds.

Table 4. Communication and computational costs.

FL Setting	Method	Comm/C/R (KB)	Comp/C/R (MB)	Rounds	Comm/C (MB)	Comp/C (GB)
Pathological CIFAR-10	FedTGP	3.91	54.52	302	1.15	16.08
	FedRAL	**1.56**	**28.23**	**201**	**0.31**	**5.54**
Pathological CIFAR-100	FedTGP	19.53	54.52	354	6.75	18.85
	FedRAL	**9.77**	**28.23**	**144**	**1.37**	**3.97**
Practical CIFAR-10	FedTGP	19.53	54.87	348	6.64	18.65
	FedRAL	**7.81**	**28.58**	**103**	**0.79**	**2.87**
Practical CIFAR-100	FedTGP	195.31	54.87	363	69.24	19.45
	FedRAL	**39.06**	**28.58**	**146**	**5.57**	**4.07**

Note: "Comm/C/R": a single client's communication cost (KB) in a communication round. "Comp/C/R": a single client's computational cost (MB) in a communication round. "Rounds": the communication rounds required to reach the specified $\{90, 50, 40, 9\}\%$ target accuracy (the near maximum accuracy that both the state-of-the-art FedTGP baseline and FedRAL can achieve). "Comm/C': a single client's communication cost (MB) required to reach the target accuracy. "Comm/C': a single client's computational cost (GB) required to reach the target accuracy.

Table 5. PSNR under DLG attacks on CIFAR-100.

Method	LG-FedAvg	FedKD	FedAPEN	FedProto	FedTGP	FedRAL
PSNR (dB)	7.83	6.95	6.93	6.76	6.69	**6.28**

Table 6. Average Accuracy (%) under high model heterogeneity. "-" denotes failure to converge.

Method	LG-FedAvg	FedKD	FedAPEN	FedProto	FedTGP	FedRAL
Accuracy (%)	45.63	45.95	-	48.56	48.69	**50.79**

signed representation angle learning facilitates model convergence. Overall, FedRAL consumes lower communication costs for reaching the target accuracy than FedTGP (column-6), demonstrating its efficient communication.

Computational Cost. Table 4 also shows the computational costs of the state-of-the-art FedTGP baseline and FedRAL. We can see that FedRAL consumes a lower single-round computational cost than FedTGP (column-4), since additionally training a homogeneous representation angle square matrix in FedRAL consumes lower FLOPs than additionally calculating label-wise average representations in FedTGP, except local model training in both methods. Since FedRAL requires fewer communication rounds to reach the specified target accuracy (column-5), its total computational costs are lower than FedTGP when reaching the specified target accuracy (column-7), substantiating its efficient computation.

Privacy Evaluation. We conduct experiments to evaluate the privacy of FedRAL and baselines. Followed by Zhang et al. [70], considering a semi-honest FL scenario where the server is untrustworthy and may inversely infer the original data of comprised clients from transmitted information by launching Deep Leakage from Gradients (DLG) attack. We use the Peak Signal-to-Noise Ratio (PSNR), a common privacy measurement metric, to evaluate the privacy-protecting level. A lower PSNR denotes better privacy protection. The results in Table 5 validate that FedRAL can protect data privacy effectively.

High Model Heterogeneity. To evaluate how FedRAL and baselines perform under high model heterogeneity, we conduct experiments CIFAR-100 with 100 clients and heterogeneous models including ResNet-{4, 6, 8, 10, 18, 34, 50, 101, 152}, CNN, and ViT. The results in Table 6 validate FedRAL's robustness to high model heterogeneity.

5.3. Robustness Analysis

This section analyzes the robustness of FedRAL to different non-IID degrees of two types.

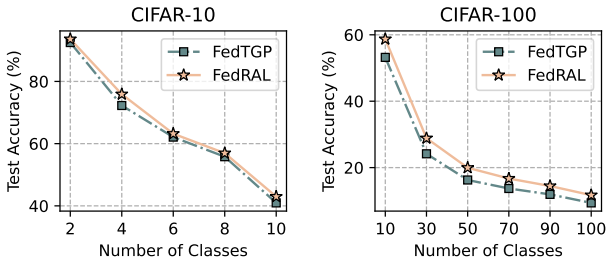

Figure 6. Robustness to pathological non-IIDness.

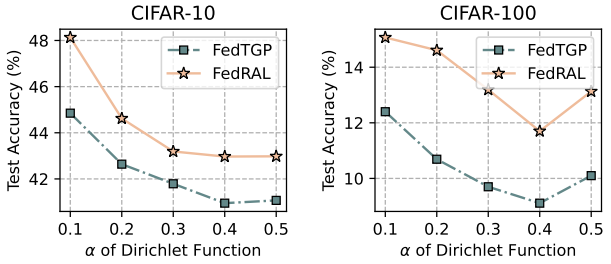

Figure 7. Robustness to practical non-IIDness.

Robustness to pathological non-IIDness. We change the number of classes allocated to each client to obtain different non-IID degrees. For CIFAR-10, we allocate $\{2, 4, 6, 8, 10\}$ classes from 10 classes to each client. For CIFAR-100, we allocate $\{10, 30, 50, 70, 90, 100\}$ classes out of 100 classes to each client. Figure 6 presents that FedRAL always performs higher accuracies than the state-of-the-art FedTGP baseline. As the number of classes assigned to one client increases, *i.e.*, the non-IID degree decreases, model accuracy tends to drop since more IID data benefits generalization while compromising personalization of local models.

Robustness to practical non-IIDness. We vary the hyperparameter α of the Dirichlet distribution function to construct diverse non-IID degrees. For both CIFAR-10 and CIFAR-100, we set $\alpha = \{0.1, 0.2, 0.3, 0.4, 0.5\}$ to evaluate the average model accuracies of FedRAL and the state-of-the-art FedTGP baseline. Figure 7 shows that FedRAL performs significantly higher average model accuracy than FedTGP, again demonstrating the robustness of FedRAL to non-IIDness. Similarly, as the value of α increases, *i.e.*, the non-IID degree decreases, model accuracy tends to degrade also due to the above reason.

5.4. Ablation Analysis

FedRAL involves three contributions: 1) introducing representation angle matrix learning to share cross-client representation angle knowledge for enhancing model generalization, 2) devising adaptive diagonal sparsification to improve communication efficiency while keeping important representation angle knowledge, 3) designing ordinate-wise weighted aggregation to fuse cross-client representation angle square matrices. Contribution-1) is essential to implement FL collaboration across clients with heterogeneous models, so we conduct experiments to evaluate the necessity of the other two contributions with three cases:

- Case-A, FedRAL with both adaptive diagonal sparsification and ordinate-wise weighted aggregation, *i.e.*, the designed method.
- Case-B, FedRAL without adaptive diagonal sparsification but with ordinate-wise weighted aggregation, *i.e.*, clients upload complete representation angle square matrices to the server.
- Case-C, FedRAL with adaptive diagonal sparsification but with naive ordinate-wise averaging aggregation, *i.e.*, the aggregation rule (a) displayed in Figure 4, ignoring client data distribution distinctions.

Table 7. Average accuracy (%) in ablation experiments.

Case	Diagonal Sparsification	Weighted Aggregation	Pathological		Practical	
			CIFAR-10	CIFAR-100	CIFAR-10	CIFAR-100
A	✓	✓	**93.60**	**58.70**	**42.97**	**11.70**
B	✗	✓	91.64	56.58	41.37	11.58
C	✓	✗	93.47	53.94	40.52	9.24

Table 7 shows that removing anyone degrades accuracy. Especially, replacing ordinate-wise weighted aggregation with naive averaging aggregation presents obvious accuracy degradations under most settings. These results reflect the effectiveness and necessity of the proposed two designs.

6. Conclusion

This work proposed FedRAL with three innovative designs: representation angle learning, adaptive diagonal sparsification, and element-wise weighted aggregation, The three designs enable enabling it to be a privacy-preserved, well-performed, communication and computation-efficient model-heterogeneous FL approach. Extensive experiments demonstrate its advantages in accuracy, communication and computation efficiency.

Acknowledgment

Xiaoguang Liu is supported by the National Science Foundation of China under Grant 62272252 & 62272253, and the Fundamental Research Funds for the Central Universities. Han Yu is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 (RG101/24); the RIE2025 Industry Alignment Fund – Industry Collaboration Projects (IAF-ICP) (Award I2301E0026), administered by A*STAR, as well as supported by Alibaba Group and NTU Singapore through Alibaba-NTU Global e-Sustainability CorpLab (ANGEL).

References

[1] Jin-Hyun Ahn et al. Wireless federated distillation for distributed edge learning with heterogeneous data. In *Proc. PIMRC*, pages 1–6, Istanbul, Turkey, 2019. IEEE. 2

[2] Jin-Hyun Ahn et al. Cooperative learning VIA federated distillation OVER fading channels. In *Proc. ICASSP*, pages 8856–8860, Barcelona, Spain, 2020. IEEE. 2

[3] Samiul Alam et al. Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction. In *Proc. NeurIPS*, virtual, 2022. . 2

[4] Sara Babakniya et al. Revisiting sparsity hunting in federated learning: Why does sparsity consensus matter? *Transactions on Machine Learning Research*, 1(1):1, 2023.

[5] Yun-Hin Chan, Rui Zhou, Running Zhao, Zhihan JIANG, and Edith C. H. Ngai. Internal cross-layer gradients for extending homogeneity to heterogeneity in federated learning. In *Proc. ICLR*, page 1, Vienna, Austria, 2024. OpenReview.net. 2

[6] Hongyan Chang et al. Cronus: Robust and heterogeneous collaborative learning with black-box knowledge transfer. In *Proc. NeurIPS Workshop*, virtual, 2021. . 2

[7] Jiangui Chen et al. Fedmatch: Federated learning over heterogeneous question answering data. In *Proc. CIKM*, pages 181–190, virtual, 2021. ACM. 2

[8] Sijie Cheng et al. Fedgems: Federated learning of larger server models via selective knowledge fusion. *CoRR*, abs/2110.11027, 2021. 2

[9] Yae Jee Cho et al. Heterogeneous ensemble knowledge transfer for training large models in federated learning. In *Proc. IJCAI*, pages 2881–2887, virtual, 2022. ijcai.org. 2

[10] Patryk Chrabaszcz, Ilya Loshchilov, and Frank Hutter. A downsampled variant of imagenet as an alternative to the CIFAR datasets. *CoRR*, abs/1707.08819, 2017. 5

[11] Liam Collins et al. Exploiting shared representations for personalized federated learning. In *Proc. ICML*, pages 2089–2099, virtual, 2021. PMLR. 1, 2

[12] Enmao Diao. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. In *Proc. ICLR*, page 1, Virtual Event, Austria, 2021. OpenReview.net. 2

[13] Tao Fan, Hanlin Gu, et al. Ten challenging problems in federated foundation models. *IEEE Transactions on Knowledge and Data Engineering*, 2025. 1

[14] Randy Goebel, Han Yu, Boi Faltings, Lixin Fan, and Zehui Xiong. *Trustworthy Federated Learning*. Springer, Cham, 2023. 1

[15] Xuan Gong et al. Federated learning via input-output collaborative distillation. In *Proc. AAAI*, pages 22058–22066, Vancouver, Canada, 2024. AAAI Press. 2

[16] Chaoyang He et al. Group knowledge transfer: Federated learning of large cnns at the edge. In *Proc. NeurIPS*, virtual, 2020. . 2

[17] S. Horváth. FjORD: Fair and accurate federated learning under heterogeneous targets with ordered dropout. In *Proc. NIPS*, pages 12876–12889, Virtual, 2021. OpenReview.net. 2

[18] Edward J. Hu et al. Lora: Low-rank adaptation of large language models. In *ICLR*, page 1, Virtual, 2022. OpenReview.net. 2

[19] Wenke Huang et al. Learn from others and be yourself in heterogeneous federated learning. In *Proc. CVPR*, pages 10133–10143, virtual, 2022. IEEE. 2

[20] Wenke Huang et al. Few-shot model agnostic federated learning. In *Proc. MM*, pages 7309–7316, Lisboa, Portugal, 2022. ACM.

[21] Sohei Itahara et al. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data. *IEEE Trans. Mob. Comput.*, 22(1):191–205, 2023. 2

[22] Jaehee Jang et al. Fedclassavg: Local representation learning for personalized federated learning on heterogeneous neural networks. In *Proc. ICPP*, pages 76:1–76:10, virtual, 2022. ACM. 2

[23] Eunjeong Jeong et al. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-iid private data. In *Proc. NeurIPS Workshop*, virtual, 2018. . 2

[24] Peter Kairouz et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021. 1

[25] Shivam Kalra et al. Decentralized federated learning through proxy model sharing. *Nature communications*, 14(1):2899, 2023. 2

[26] Alex Krizhevsky et al. *Learning multiple layers of features from tiny images*. Toronto, ON, Canada, , 2009. 5

[27] Meng Lei, Qi Zhuang, Wu Lei, Du Xiaoyu, Li Zhaochuan, Cui Lizhen, and Meng Xiangxu. Improving global generalization and local personalization for federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36, 2024. 1

[28] Daliang Li and Junpu Wang. Fedmd: Heterogenous federated learning via model distillation. In *Proc. NeurIPS Workshop*, virtual, 2019. . 1, 2

[29] Qinbin Li et al. Practical one-shot federated learning for cross-silo setting. In *Proc. IJCAI*, pages 1484–1490, virtual, 2021. ijcai.org. 2

[30] Paul Pu Liang et al. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 1(1), 2020. 1, 2, 6, 7

[31] Tao Lin et al. Ensemble distillation for robust model fusion in federated learning. In *Proc. NeurIPS*, virtual, 2020. . 1, 2

[32] Chang Liu et al. Completely heterogeneous federated learning. *CoRR*, abs/2210.15865, 2022. 2

[33] Qianyu Long, Christos Anagnostopoulos, Shameem Puthiya Parambath, and Daning Bi. Feddip: Federated learning with extreme dynamic pruning and incremental regularization. In *Proc.ICDM*, pages 1187–1192, Shanghai, China, 2023. IEEE. 2

[34] Disha Makhija et al. Architecture agnostic federated learning for neural networks. In *Proc. ICML*, pages 14860–14870, virtual, 2022. PMLR. 2

[35] Koji Matsuda et al. Fedme: Federated learning via model exchange. In *Proc. SDM*, pages 459–467, Alexandria, VA, USA, 2022. SIAM. 2

[36] Brendan McMahan et al. Communication-efficient learning of deep networks from decentralized data. In *Proc. AISTATS*, pages 1273–1282, Fort Lauderdale, FL, USA, 2017. PMLR. 1

[37] Duy Phuong Nguyen et al. Enhancing heterogeneous federated learning with knowledge extraction and multi-model fusion. In *Proc. SC Workshop*, pages 36–43, Denver, CO, USA, 2023. ACM. 2

[38] Jaehoon Oh et al. Fedbabu: Toward enhanced representation for federated image classification. In *Proc. ICLR*, virtual, 2022. OpenReview.net. 2

[39] Zhengxiang Pan, Han Yu, Chunyan Miao, and Cyril Leung. Efficient collaborative crowdsourcing. In *The 30th AAAI Conference on Artificial Intelligence (AAAI-16)*, pages 4248–4249, 2016. 1

[40] Sejun Park et al. Towards understanding ensemble distillation in federated learning. In *Proc. ICML*, pages 27132–27187, Honolulu, Hawaii, USA, 2023. PMLR. 2

[41] Krishna Pillutla et al. Federated learning with partial model personalization. In *Proc. ICML*, pages 17716–17758, virtual, 2022. PMLR. 2

[42] Zhuang Qi, Sijin Zhou, Lei Meng, Han Yu, and Xiangxu Meng. Federated deconfounding and debiasing learning for out-of-distribution generalization. In *The 34th International Joint Conference on Artificial Intelligence, (IJCAI'25)*, pages 1–9, 2025. 1

[43] Zhen Qin et al. Fedapen: Personalized cross-silo federated learning with adaptability to statistical heterogeneity. In *Proc. KDD*, pages 1954–1964, Long Beach, CA, USA, 2023. ACM. 2, 6, 7

[44] Zeju Qiu et al. Controlling text-to-image diffusion by orthogonal finetuning. In *Proc. NeurIPS*, New Orleans, LA, USA, 2023. 2, 3

[45] Chao Ren, Han Yu, et al. Advances and open challenges in federated foundation models. *IEEE Communications Surveys and Tutorials*, 2025. 1

[46] Felix Sattler et al. Fedaux: Leveraging unlabeled auxiliary data in federated learning. *IEEE Trans. Neural Networks Learn. Syst.*, 1(1):1–13, 2021. 2

[47] Felix Sattler et al. CFD: communication-efficient federated distillation via soft-label quantization and delta coding. *IEEE Trans. Netw. Sci. Eng.*, 9(4):2025–2038, 2022. 2

[48] Aviv Shamsian et al. Personalized federated learning using hypernetworks. In *Proc. ICML*, pages 9489–9502, virtual, 2021. PMLR. 6

[49] Tao Shen et al. Federated mutual learning. *CoRR*, abs/2006.16765, 2020. 2

[50] Alysa Ziying Tan et al. Towards personalized federated learning. *IEEE Trans. Neural Networks Learn. Syst.*, 1(1): 1–17, 2022. 1

[51] Yue Tan et al. Fedproto: Federated prototype learning across heterogeneous clients. In *Proc. AAAI*, pages 8432–8440, virtual, 2022. AAAI Press. 2, 6, 7

[52] Jiaqi Wang et al. Towards personalized federated learning via heterogeneous model reassembly. In *Proc. NeurIPS*, page 13, New Orleans, Louisiana, USA, 2023. OpenReview.net. 2

[53] Chuhan Wu et al. Communication-efficient federated learning via knowledge distillation. *Nature Communications*, 13 (1):2032, 2022. 2, 6, 7

[54] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017. 5

[55] Qiang Yang, Lixin Fan, and Han Yu. *Federated Learning: Privacy and Incentive*. Springer, Cham, 2020. 1

[56] Qiang Yang, Yang Liu, Yong Cheng, Yan Kang, Tianjian Chen, and Han Yu. *Federated Learning*. Springer, Cham, 2020. 1

[57] Liping Yi, Gang Wang, and Xiaoguang Liu. QSFL: A two-level uplink communication optimization framework for federated learning. In *Proc. ICML*, pages 25501–25513. PMLR, 2022. 1

[58] Liping Yi, Xiaorong Shi, Nan Wang, Gang Wang, Xiaoguang Liu, Zhuan Shi, and Han Yu. pfedkt: Personalized federated learning with dual knowledge transfer. *Knowledge-Based Systems*, 292:111633, 2024. 1

[59] Liping Yi, Xiaorong Shi, Nan Wang, Jinsong Zhang, Gang Wang, and Xiaoguang Liu. Fedpe: Adaptive model pruning-expanding for federated learning on mobile devices. *IEEE Transactions on Mobile Computing*, pages 1–18, 2024. 1

[60] Liping Yi, Gang Wang, Xiaofei Wang, and Xiaoguang Liu. Qsfl: Two-level communication-efficient federated learning on mobile edge devices. *IEEE Transactions on Services Computing*, pages 1–16, 2024. 1

[61] Liping Yi, Han Yu, Zhuan Shi, Gang Wang, Xiaoguang Liu, Lizhen Cui, and Xiaoxiao Li. FedSSA: Semantic Similarity-based Aggregation for Efficient Model-Heterogeneous Personalized Federated Learning. In *IJCAI*, 2024. 1

[62] Liping Yi et al. FedGH: Heterogeneous federated learning with generalized global header. In *Proc. ACM MM*, 2023. 2, 6

[63] Liping Yi et al. Federated model heterogeneous matryoshka representation learning. In *Proc. NeurIPS*, Vancouver, Canada, 2024. .

[64] Liping Yi et al. pFedAFM: Adaptive Feature Mixture for Data-Level Personalization in Heterogeneous Federated Learning on Mobile Edge Devices . In *Proc. ICDE*, pages 1981–1994, Hong Kong SAR, China, 2025. IEEE Computer Society.

[65] Liping Yi et al. pfedes: Generalized proxy feature extractor sharing for model heterogeneous personalized federated learning. In *Proc. AAAI*, pages 22146–22154, Philadelphia, PA, USA, 2025. AAAI Press. 1

[66] Fuxun Yu et al. Fed2: Feature-aligned federated learning. In *Proc. KDD*, pages 2066–2074, virtual, 2021. ACM. 2

[67] Han Yu, Siyuan Liu, Alex C Kot, Chunyan Miao, and Cyril Leung. Dynamic witness selection for trustworthy distributed cooperative sensing in cognitive radio networks. In *Proceedings of the 13th IEEE International Conference on Communication Technology (ICCT'11)*, pages 1–6, 2011. 1

[68] Sixing Yu et al. Resource-aware federated learning using knowledge extraction and multi-model fusion. *CoRR*, abs/2208.07978, 2022. 2

[69] Jie Zhang et al. Parameterized knowledge transfer for personalized federated learning. In *Proc. NeurIPS*, pages 10092–10104, virtual, 2021. OpenReview.net. 2

[70] Jianqing Zhang et al. Fedcp: Separating feature information for personalized federated learning via conditional policy. In *Proc. KDD*, page 1, Long Beach, CA, USA, 2023. ACM. 7

[71] Jie Zhang et al. Towards data-independent knowledge transfer in model-heterogeneous federated learning. *IEEE Trans. Computers*, 72(10):2888–2901, 2023. 2

[72] Jianqing Zhang et al. Fedtgp: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning. In *Proc. AAAI*, pages 16768–16776, Vancouver, Canada, 2024. 1. 6, 7

[73] Lan Zhang et al. Fedzkt: Zero-shot knowledge transfer towards resource-constrained federated learning with heterogeneous on-device models. In *Proc. ICDCS*, pages 928–938, virtual, 2022. IEEE. 2

[74] Zhilu Zhang and Mert R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proc. NeurIPS*, pages 8792–8802, Montréal, Canada, 2018. Curran Associates Inc. 4

[75] Hangyu Zhu et al. Federated learning on non-iid data: A survey. *Neurocomputing*, 465:371–390, 2021. 1

[76] Zhuangdi Zhu et al. Data-free knowledge distillation for heterogeneous federated learning. In *Proc. ICML*, pages 12878–12889, virtual, 2021. PMLR. 2

[77] Zhuangdi Zhu et al. Resilient and communication efficient learning for heterogeneous federated systems. In *Proc. ICML*, pages 27504–27526, virtual, 2022. PMLR. 2

[78] Qi Zhuang, Meng Lei, Chen Zitan, Hu Han, Lin Hui, and Meng Xiangxu. Cross-silo prototypical calibration for federated learning with non-iid data. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3099–3107, 2023. 1

[79] Qi Zhuang, Meng Lei, et al. Cross-silo feature space alignment for federated learning on clients with imbalanced data. In *The 39th Annual AAAI Conference on Artificial Intelligence (AAAI-25)*, pages 19986–19994, 2025. 1