Re-architecting Personalized Federated Learning for Demanding Edge Environments

Quyang Pan[1,2], Sheng Sun[1], Tingting Wu[3], Zhiyuan Wu[1,2], Yuwei Wang[1*], Min Liu[1], Bo Gao[4], Jingyuan Wang[5]

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100190, China
[3]China Mobile Research Institute, Xicheng, Beijing 100053, China
[4]School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China
[5]MOE Engineering Research Center of Advanced Computer Application Technology, SCSE, Beihang University, China
{panquyang23s, sunsheng, wuzhiyuan22s, ywwang, liumin}@ict.ac.cn;
wutingtingyjy@chinamobile.com; bogao@bjtu.edu.cn; jywang@buaa.edu.cn

Abstract

Federated Edge Learning (FEL) has emerged as a promising approach for enabling edge devices to collaboratively train machine learning models while preserving data privacy. Despite its advantages, practical FEL deployment faces significant challenges related to device constraints and device-server interactions, necessitating heterogeneous, user-adaptive model training with limited and uncertain communication. While knowledge cache-driven federated learning offers a promising FEL solution for demanding edge environments, its logits-based interaction design provides poor richness of exchanged information for on-device model optimization. To tackle this issue, we introduce DistilCacheFL, a novel personalized FEL architecture that enhances the exchange of optimization insights while delivering state-of-the-art performance with efficient communication.

DistilCacheFL incorporates the benefits of both dataset distillation and knowledge cache-driven federated learning by storing and organizing distilled data as knowledge in the server-side knowledge cache, allowing devices to periodically download and utilize personalized knowledge for local model optimization. Moreover, a device-centric cache sampling strategy is introduced to tailor transferred knowledge for individual devices within controlled communication bandwidth. Extensive experiments on five datasets covering image recognition, audio understanding, and mobile sensor data mining tasks demonstrate that (1) DistilCacheFL significantly outperforms state-of-the-art methods regardless of model structures, data distributions, and modalities. (2) DistilCacheFL can train splendid personalized on-device models with at least $\times$28.6 improvement in communication efficiency.

Introduction

Federated Edge Learning (FEL) (Tak and Cherkaoui 2021; Duan et al. 2023) is a specialized form of Federated Learning (Yang et al. 2019) designed to operate at the edge of the network. It enables edge devices (clients) to jointly train machine learning models under the coordination of an edge server (server) without sharing raw data. With the growing prevalence of mobile and Internet of Things (IoT) devices coupled with increasing concerns over data privacy, FEL has empowered wide adoption of various on-device Artificial Intelligence (AI) applications, including smart transportation (Wang, Lin, and Li 2025), healthcare (Nguyen et al. 2022), and recommendation (Yuan et al. 2023; Guo et al. 2021).

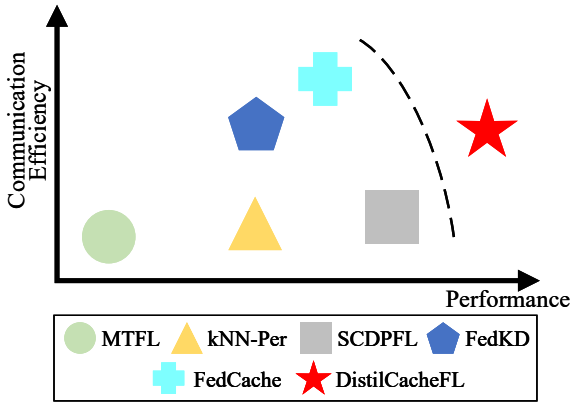


Figure 1: Comparison of DistilCacheFL with state-of-the-art methods.

Despite its promising potential, FEL faces significant challenges in practical deployment. One of the primary challenges is the diversity of edge device infrastructures. Devices such as smartwatches, mobile phones, and tablets differ significantly in terms of computational power, storage capacity, and battery life (Tak and Cherkaoui 2021; Yu and Li 2021). This diversity calls for the deployment of highly scalable, user-adaptive models that can accommodate heterogeneous hardware specifications across devices (Tak and Cherkaoui 2021; Duan et al. 2023; Tan et al. 2023; Yu and Li 2021). Even among devices with similar hardware configurations, data distributions and user preferences are often highly individualized, further complicating model generalization (Tan et al. 2022; Mills, Hu, and Min 2022; Wu et al. 2023b). Another challenge arises from the communication limitations in mobile edge networks. Edge devices frequently operate

*Corresponding Author.

under low bandwidth and unreliable network conditions (Tak and Cherkaoui 2021; Zhang et al. 2022; Zhu et al. 2022), making the transmission of large-scale model parameters impractical due to the precious nature of wireless channels (Al-Quraan et al. 2023; Wu et al. 2022). Moreover, devices are often intermittently online (Zhu et al. 2022), further complicating the coordination required for collaborative model updates. The aforementioned constraints hinder the efficiency of the FEL process, ultimately impacting the systems' overall performance and the benefits it delivers to users.

Numerous studies have sought to overcome the aforementioned challenges in FEL. Heterogeneous FL (Zhu, Hong, and Zhou 2021; Li and Wang 2019; Wu et al. 2024b) focuses on enabling collaborative training across devices with different model architectures, allowing adaptation to varying computational resources and hardware capabilities. Personalized or multi-task FL (Mills, Hu, and Min 2022; Marfoq et al. 2022; Jin et al. 2022) centers on developing models tailored to individual devices, accommodating diverse user preferences. Communication-efficient FL (Wu et al. 2022; Sattler et al. 2021) typically incorporates techniques such as model compression or quantization, reducing the training communication burden with little performance loss. However, each of these approaches addresses only a single challenge, limiting their applicability in real-world edge deployments where multiple challenges coexist.

Given the complexity of the aforementioned challenges, it is evident that a multi-faceted FEL solution is urgently needed. Fortunately, knowledge cache-driven FL (FedCache) (Wu et al. 2024c) offers a prevailing paradigm that revolutionizes the mainstream parameters interaction protocol (McMahan et al. 2017; Li et al. 2020; Reddi et al. 2021; Chen et al. 2023a; Mills, Hu, and Min 2022; Jin et al. 2022; Lee et al. 2022; Wu et al. 2024a) in FEL. By performing on-device personalized distillation driven by the self-organizing server-side knowledge cache, FedCache facilitates communication-efficient and heterogeneous-compatible collaborative training without requiring multiple devices to remain online simultaneously. However, the performance of FedCache is limited by the logits interaction design, which restricts the amount and quality of information that can be interacted between devices and the server. In addition, the applicability of FedCache across various data modalities and application tasks is also restricted due to its reliance on task-specific encoders.

In this paper, we introduce DistilCacheFL, a novel personalized FEL architecture that improves the performance of heterogeneous on-device models with efficient communication and uncertain connection tolerance, as displayed in Fig. 1. Specifically, DistilCacheFL revolute the transferred knowledge by shifting from logits to distilled data (Lei and Tao 2023), offering a novel interaction paradigm between devices and the server. In our novel design, devices perform dataset distillation with the assistance of cached knowledge from the remote server. The distilled data is then shared with the server, ensuring the knowledge cache remains updated with the latest information. To balance system performance and communication efficiency, a device-centric cache sampling strategy is proposed for tailoring transferred knowledge for individual devices within the constraints of available communication bandwidth. The key superiorities of DistilCacheFL compared with the original FedCache are twofold. First, DistilCacheFL provides richer information characterization capabilities by storing and transferring distilled synthetic data rather than logits, enabling on-device models to optimize with sufficient server-side information and achieve better precision. Second, DistilCacheFL adopts a more generalized data anonymization method, enhancing its extensibility to a broader range of data modalities and application tasks. Our proposed architecture maintains the advantages of model heterogeneity allowance, learning personalization, uncertain connection tolerance, and efficient communication from FedCache, while also achieving remarkable performance gains by fully exploiting the knowledge from distilled data.

Contributions. The main contributions of this paper are as follows: (1) We propose DistilCacheFL, a novel personalized FEL architecture that integrates dataset distillation with knowledge cache-driven federated learning, aiming to achieve state-of-the-art performance while tackling the multi-faceted challenges of edge deployment. (2) We introduce federated dataset distillation and device-centric cache sampling that matches DistilCacheFL design, facilitating knowledge generation, storage, and organization as well as personalized model training, with data privacy protected. (3) We conduct comprehensive experiments on five datasets, encompassing image recognition, audio understanding, and mobile sensor data mining tasks. Built upon diversified data heterogeneity, model settings, and application scenarios, DistilCacheFL not only consistently outperforms state-of-the-art methods (at least 1.7% average User model Accuracy (Mills, Hu, and Min 2022) enhancement) in all considered settings, but also achieves better communication efficiency (at least $\times 29.6$) compared with baseline algorithms.

Related Work

Personalized Federated Learning. A variety of approaches have been developed to tackle the dual challenges of learning and model scale personalization within FL. Differentiated client-side model optimization objectives are implemented in studies such as (Marfoq et al. 2022; Shi et al. 2023; Mills, Hu, and Min 2022; Jin et al. 2022; Chen et al. 2023a; Yang et al. 2023; Xue et al. 2025b), enabling trained models to generalize across clients with varying local data distributions. Novel client-server interaction designs, which depart from the traditional FedAvg (McMahan et al. 2017), are explored in (Wu et al. 2022; Xue et al. 2025a; Zhu, Hong, and Zhou 2021; Wu et al. 2024b; Huang, Ye, and Du 2022) to better accommodate diverse client hardware configurations with differently structured models. Furthermore, hybrid approaches such as (Wu et al. 2024c, 2023b) are proposed to simultaneously address both model and learning personalization in FL.

Federated Learning in Edge Computing. The efficiency of executing FL at the network edge has become a hot topic. Research works such as (Alam et al. 2022; Cho et al. 2022; He, Annavaram, and Avestimehr 2020; Wen et al. 2024) investigate the technical frameworks required for running FL algorithms on devices constrained by computational power or storage resources. Device heterogeneity and connection

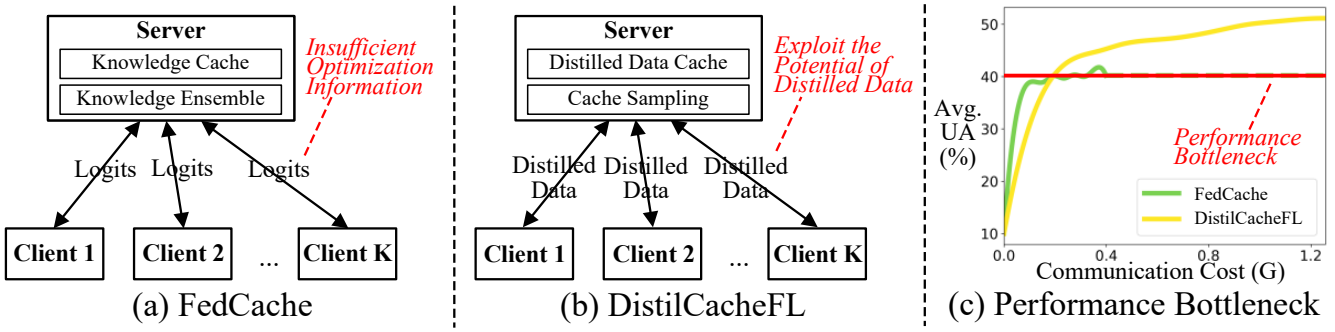


Figure 2: Comparison of FedCache and DistilCacheFL. Results in (c) are derived on the CIFAR-10 dataset, taking $\alpha = 0.5$ and $K = 100$.

uncertainty in edge environments are tackled by methodologies such as (Zhu et al. 2022; Wu et al. 2024c; Liu et al. 2023; Wu et al. 2023b). Moreover, (Wang et al. 2022; Wu et al. 2023a; Deng et al. 2023; Liu et al. 2020; Wang et al. 2021) extends FL to a multi-tier architecture involving end-edge-cloud collaborations, enhancing model training efficiency and final performance by leveraging the edge as a bridge between devices and the cloud during the training process.

Federated Learning with Alternative Information. Instead of transmitting model parameters, alternative information is utilized in the FL training process by a series of recent works. Model-agnostic outputs are exchanged between clients and the server in (Wu et al. 2024c; Itahara et al. 2023; Huang, Ye, and Du 2022; Wu et al. 2024b; Li and Wang 2019), allowing deployment of customized on-device models across resource-heterogeneous clients. Additionally, methodologies involving uploading mixed or distilled data from clients to servers are proposed in (Song et al. 2023; Oh et al. 2020), significantly reducing communication overhead while maintaining client data privacy.

Problem Statement and Reformulation

Background and Preliminary. We consider an FL system deployed at the edge of the network, comprising K participating edge devices (clients) coordinated by an edge server (server). Each client $k \in \{1, 2, ..., K\}$ owns its local dataset $\mathcal{D}^k = \bigcup_{i=1}^{|\mathcal{D}^k|} \{(X_i^k, y_i^k)\}$ with $|\mathcal{D}^k|$ samples, where each sample are with D data dimensions and belong to one of C distinct classes. Due to differentiated user behaviors, both the local training an testing datasets among clients are non-independently and identically distributed. Throughout this paper, the terms 'device' and 'client' are used interchangeably. Assume that the personalized model parameters of client k are denoted as $W^k \in \mathbb{R}^{d^k}$, where d^k indicates the number of parameters in the model of client k. Due to system heterogeneity among devices, the required model sizes may vary across clients, such that $d_l \neq d_m, \exists l, m \in \{1, 2, ..., K\}$. Each client k has a local objective $\mathcal{L}^i : \mathbb{R}^D \to \mathbb{R}$, which relies on its corresponding local data distribution. The overall goal is to minimize the expected objective across all clients, which is formally expressed as:

$$\min_{\bigcup_{k=1}^{K}\{W^k\}} \frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{|\mathcal{D}^k|}\sum_{(X_i^k,y_i^k)\in\mathcal{D}^k}\mathcal{L}^i(W^k; X_i^k, y_i^k)\right). \tag{1}$$

Given the instability of device connections in edge environments, multiple clients may not be online simultaneously. Besides, it is essential to minimize the communication overhead between devices and the server under the premise of guaranteeing user model accuracy (Mills, Hu, and Min 2022), saving valuable wireless network resources as well as device energy.

Knowledge Cache-driven Federated Learning. We formulate knowledge cache-driven FL as a distributed optimization problem with the assistance of the remote knowledge cache KC on the server, that is:

$$\min_{\bigcup_{k=1}^{K}\{W^k\}} \frac{1}{K}\sum_{k=1}^{K}\Big(\frac{1}{|\mathcal{D}^k|}\Big(\sum_{(X_i^k,y_i^k)\in\mathcal{D}^k}\mathcal{L}_{CE}(W^k; \varphi(F^k(X_i^k)), y_i^k) + \beta\cdot\mathcal{R}^k(W^k; KC)\Big)\Big), \tag{2}$$

where $\mathcal{L}_{CE}$ is the cross-entropy loss, φ is the softmax function, F^k is the prediction function of the model on client k. $\mathcal{R}^k$ represents the redundant optimization component of client k based on cached knowledge, with corresponding weighting term β. As an example, FedCache (Wang and Yoon 2021) considers model outputs (logits) as knowledge, and optimizes local models based on cached related knowledge, that is:

$$\mathcal{R}^k = \sum_{(X_i^k,y_i^k)\in\mathcal{D}^k}\mathcal{L}_{KL}(\varphi(F^k(X_i^k))||\varphi(\frac{1}{R}\sum_{(zr_i^k)_s\in KC[k,i]}(zr_i^k)_s)), \tag{3}$$

where $\mathcal{L}_{KL}$ is the Kullback-Leibler Divergence loss, $(zr_i^k)_s$ is the s-th knowledge fetched from the knowledge cache for sample index (k, i), R is a hyper-parameter that controls the number of related knowledge in FedCache. However, FedCache exhibits severe limitations in providing rich, distribution-aware information for personalized optimization over devices. The amount of information attainable from the remote knowledge cache is significantly restricted due to the design of small-scale logits interactions, as shown in Figure

2 (a). This design fails to offer sufficient optimization information for clients, leading to performance bottlenecks of FedCache, as shown in Figure 2 (c). Additionally, FedCache relies on task-specific data encoders to capture private sample relations, which restricts its applicability across varied data modalities and application tasks.

DistilCacheFL Optimization Formulation. To address the aforementioned shortcomings of FedCache, DistilCacheFL is designed to revolutionize the transferred knowledge by shifting from logits to distilled data, as shown in Figure 2 (b). Specifically, local model optimization in DistilCacheFL is regulated by post-sampled distilled data jointly synthesized by clients, that is:

$$\mathcal{R}^k = \sum_{(X^*,y^*)\in Sub^k(\bigcup_{l=1}^{K}\hat{\mathcal{D}}^l_{distill})} \mathcal{L}_{CE}(W^k;\varphi(F^k(X^*)),y^*), \tag{4}$$

where $\hat{\mathcal{D}}^l_{distill}$ is the synthetic data distilled on client l, Sub^k represents the adaptive sample strategy tailered for client k. In our design, the synthesized data after sampling serves as the knowledge that devices request from the knowledge cache. This reformulation not only provides more comprehensive semantic information for local training on clients but also enhances the control over downloaded cached knowledge, enabling task-compatible and communication-efficient personalized optimization.

DistilCacheFL

In this section, we introduce our proposed DistilCacheFL with an overview illustrated in Figure 3. An execution procedure of DistilCacheFL is elaborated in Algorithm 1.

Knowledge Cache Design

Building upon the principles of knowledge-driven FL, DistilCacheFL caches the latest distilled data as knowledge on the server side. In terms of knowledge cache operations, we provide two operations for indexing knowledge in the cache.

Client-Based Indexing. Each client's distilled data is indexed by their identifier, allowing for efficient updates of knowledge in the cache and prototype initialization for on-device distillation, that is:

$$KC[client,k] \leftarrow \hat{\mathcal{D}}^k_{distill}, \forall k \in \{1,2,...,K\}, \tag{5}$$

where KC is the notation of the knowledge cache on the server.

Class-Based Indexing. All cached knowledge belonging to any specific class $y^* \in \{1,2,...,C\}$ are jointly fetched, facilitating the subsequent device-centric client sampling process, that is:

$$S_c \leftarrow KC[class,c], \forall c \in \{1,2,...,C\}, \tag{6}$$

where S_c the set of all knowledge belong to class c in the knowledge cache, subject to:

$$\begin{aligned} &S_c = \{(X^*,y^*)|(X^*,y^*) \in KC[client,k],\\ &k \in \{1,2,...,K\}, y^* = c\}. \end{aligned} \tag{7}$$

Federated Dataset Distillation

DistilCacheFL introduces federated dataset distillation, which collaboratively extracts anonymous structured information from local data on individual clients. This distilled data is stored on the server for further organization and accessibility.

On-Device Dataset Distillation. All devices decompose their local models into feature extractors and classifiers. For each given sample $(X^*,y^*) \in \mathcal{D}^k$ on device k, the outputs of corresponding feature extractors and classifiers are denoted as $F^k_f(X^*)$ and $F^k_c(F^k_f(X^*))$, respectively. To start dataset distillation, each device k initializes its prototype by selecting one local sample per class during the first communication round or receiving distilled data from other clients during subsequent communication rounds. The latter process is controlled by a periodically updated random replacement function $\sigma : \{1,2,...,K\} \rightarrow \{1,2,...,K\}$, with the intermediate distilled data stored in the knowledge cache, that is:

$$\hat{\mathcal{D}}^k_b \leftarrow \begin{cases} KC[client,\sigma(k)], & KC[client,\sigma(k)] \neq \phi \\ \mathcal{D}^k_0, & KC[client,\sigma(k)] = \phi \end{cases}, \tag{8}$$

where $\hat{\mathcal{D}}^k_b$ denotes the set of prototype samples to be optimized into synthetic data after distillation. $\mathcal{D}^k_0$ is a subset of $\mathcal{D}^k$ with C elements, subject to:

$$\begin{aligned} &y^k_0 \neq y^{k\prime}_0 \vee y^k_0 = y^{k\prime}_0 \wedge X^k_0 = X^{k\prime}_0,\\ &\forall (X^k_0,y^k_0) \in \mathcal{D}^k_0 \wedge (X^{k\prime}_0,y^{k\prime}_0) \in \mathcal{D}^k_0. \end{aligned} \tag{9}$$

Without loss of generality, we assume device k sets up a prototype $(X^k_b,y^k_b) \in \hat{\mathcal{D}}^k_b$ on class y^k_b. The on-device dataset distillation process should include computing the distance between the prototype's feature maps and those of the local data using the Gram matrix, that is:

$$K^k_{bl} = F^k_f(X^k_i) \cdot F^k_f(X^k_b)^{\mathrm{T}}. \tag{10}$$

Similarly, we compute the Gram matrix of the prototype itself:

$$K^k_{bb} = F^k_f(X^k_b) \cdot F^k_f(X^k_b)^{\mathrm{T}}. \tag{11}$$

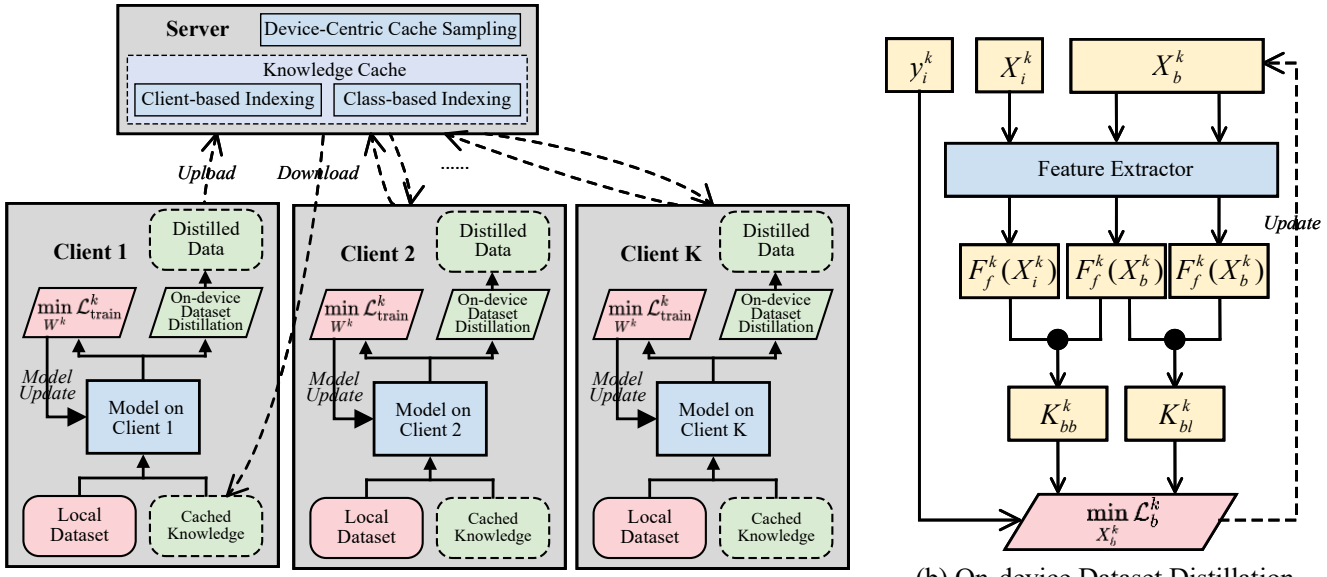
The dataset distillation objective $\mathcal{L}^k_b$ is then optimized following kernel ridge regression loss:

$$\min_{X^k_b} \mathcal{L}^k_b = \min_{X^k_b} \frac{1}{2}||y^k_b - K^k_{bl}(K^k_{bb}+\lambda I)^{-1} \cdot y^k_i||^2, \tag{12}$$

where I denotes the identity matrix, and λ is a hyperparameter to control the degree of regularization. Note that local data is often augmented using common dataset enhancement techniques to increase the diversity of local feature maps during distillation. After obtaining the distilled data on client k, it is stored in the knowledge cache KC, ensuring the devices always have access to the latest distilled knowledge in the following communication rounds, that is:

$$KC[client,k] \leftarrow \hat{\mathcal{D}}^k_b. \tag{13}$$

Collaborative Training. On-device dataset distillation relies on well-optimized feature extractors. To enhance local model performance and improve future distillation quality, devices periodically request cached distilled data from the



(a) Framework of DistilCacheFL

(b) On-device Dataset Distillation on Client k

Figure 3: Overview of DistilCacheFL.

Algorithm 1: DistilCacheFL.

procedure ServerExecute()
  // Initialization Process
  foreach client $k \in \{1, 2, \ldots, K\}$:
    $KC[client, k] \leftarrow \phi$
    foreach class $c \in \{1, 2, \ldots, C\}$:
      Receive p_c^k from client k
  // Training Process
  foreach client $k \in \{1, 2, \ldots, K\}$:
    Send possible $\hat{\mathcal{D}}_b^k$ following Eq. (8)
    Receive distilled data $\hat{\mathcal{D}}_b^k$ from client k
    Update KC following Eq. (13)
    Sample cache following Eq. (17)
    Send sampled knowledge to client k
end procedure

procedure ClientExecute(k)
  // Initialization Process
  foreach class $c \in \{1, 2, \ldots, C\}$:
    Compute p_c^k following Eq. (16)
    Send p_c^k to the server
  // Training Process
  Initialize $\hat{\mathcal{D}}_b^k$ following Eq. (8)
  Compute K_{bl}^k following Eq. (10)
  Compute K_{bb}^k following Eq. (11)
  Optimize $\mathcal{L}_b^k$ following Eq. (12)
  Upload distilled data $\hat{\mathcal{D}}_b^k$ to server
  Receive sampled knowledge from server
  Optimize $\mathcal{L}_{train}^k$ following Eqs. (14,15)
end procedure

server for personalized optimization. This collaborative training procedure is formulated as follows:

$$\begin{aligned}&\min_{W^k} \mathcal{L}_{train}^k \\ =&\min_{W^k} \sum_{(X_i^k, y_i^k)\in\mathcal{D}^k} \mathcal{L}_{CE}(W^k; \varphi(F^k(X_i^k)), y_i^k) \\ &+g(\sum_{(X^*,y^*)\in Sub^k(\bigcup_{l=1}^{L} KC[client,l])} \mathcal{L}_{CE}(W^k; \varphi(F^k(X^*)), y^*)),\end{aligned} \tag{14}$$

where $\mathcal{L}_{train}^k$ denotes the local training loss function on client k, g is a gating function acting as an identity mapping when the knowledge cache is empty in the first communication round and resulting in 0 otherwise, that is:

$$g(x) = \begin{cases} x, & KC[client, k] \neq \phi \\ 0, & KC[client, k] = \phi \end{cases}, \forall x. \tag{15}$$

Device-Centric Cache Sampling

To enhance personalized performance while reducing communication overhead, we propose a device-centric cache sampling strategy that considers local data characteristics and communication budgets.

Local Label Distribution Computation. During the initialization process, each client k computes its local label distribution according to its label frequency, that is:

$$p_c^k = \frac{|\{(X_i^k, y_i^k) | (X_i^k, y_i^k) \in \mathcal{D}^k, y_i^k = c\}|}{|\mathcal{D}^k|}, \tag{16}$$

where p_c^k represents the label frequency of class c on client k.

Distribution-Aware Controllable Sampling. During the training process, the knowledge cache samples and distributes

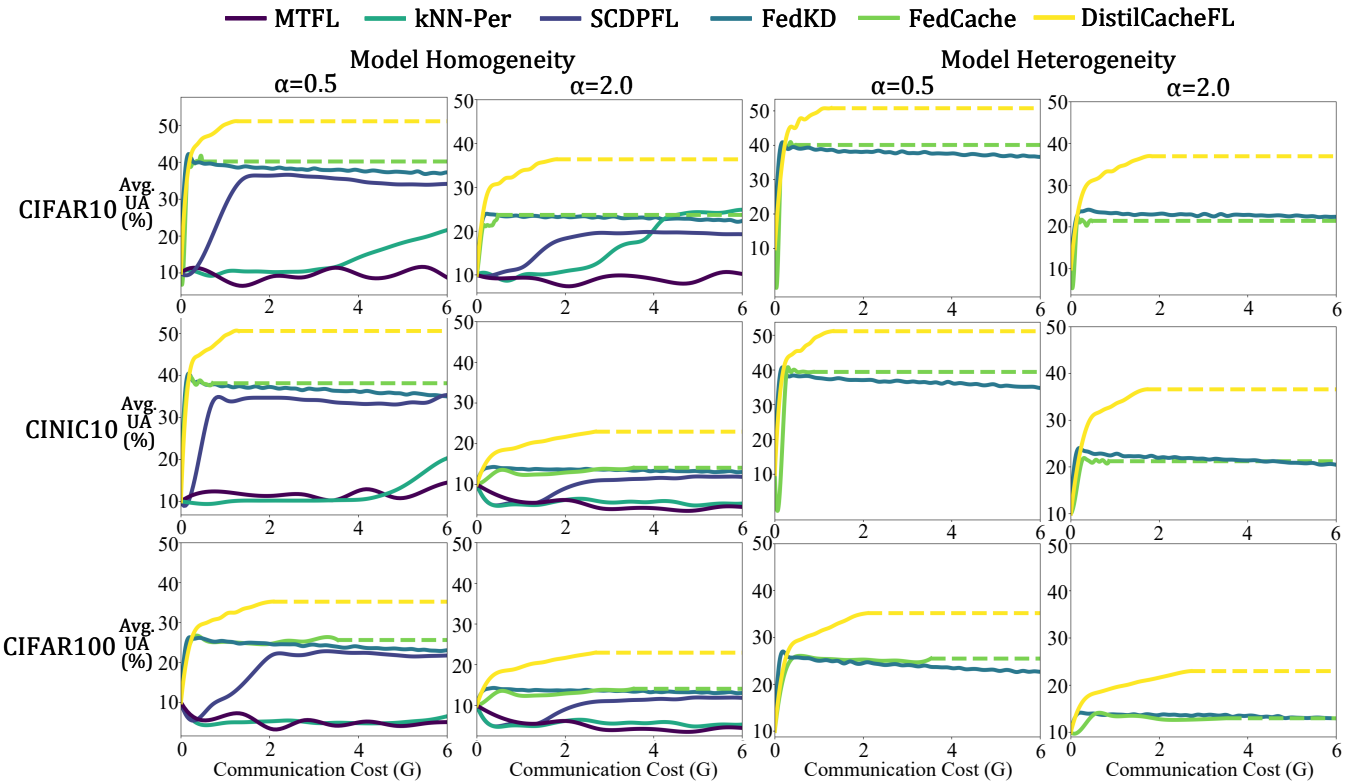


Figure 4: Average UA per unit of communication cost over image recognition tasks.

its stored knowledge based on p_c^k, that is:

$$Sub^k(\bigcup_{l=1}^{L} KC[client, l]) = \bigcup_{c=1}^{C} RS(KC[class, c], (\tau + (1-\tau)\cdot p_c^k)\cdot |KC[class, c]|), \tag{17}$$

where $RS(\hat{\mathcal{D}}^*, p_0)$ denotes random sampling in the cached knowledge set $\hat{\mathcal{D}}^*$ at a probability p_0. τ is a hyper-parameter ranging from 0 to 1 to control the trade-off between model performance and communication. As τ grows, the proportion of cached samples increases as well, leading to more cached knowledge but higher communication overhead.

Experiments

Experimental Setup

Platform. Our experiments are conducted on a high-performance physical server equipped with 12th Gen Intel(R) Core(TM) i7-12700 CPU and multiple NVIDIA GeForce RTX 3090 GPU cards. The server's memory consists of four 16GB Acer DDR4 modules operating at 2133 MT/s, providing a total of 64GB of RAM. Storage is handled by a KINGSTON SKC3000D2048G solid-state drive.

Datasets. We evaluate the effectiveness of our proposed DistilCacheFL across various application tasks, including image recognition, audio understanding, and mobile sensor data mining (Carpineti et al. 2018). These experiments cover five datasets: which are CIFAR10, CIFAR100 (Krizhevsky, Hinton et al. 2009), CINIC10 (Darlow et al. 2018), UrbanSound8K (Salamon, Jacoby, and Bello 2014), and TMD (Carpineti et al. 2018). Each complete dataset is preprocessed using the distributed data partition strategy from FedML (He et al. 2020), with a hyper-parameter α to adjust the degree of data heterogeneity among clients.

Models. We employ five model structures, considering both deep residual network (He et al. 2016) for image data, and fully connected network for numeric data.

Baselines. We compare DistilCacheFL against the following state-of-the-art methods: MTFL (Mills, Hu, and Min 2022), KNN-Per (Marfoq et al. 2022), spectral co-distillation for personalized FL (SCDPFL) (Chen et al. 2023b), FedKD (Wu et al. 2022) and FedCache (Wu et al. 2024c). These baseline algorithms encompass personalized/multi-task FL methods, FL algorithms addressing dual model heterogeneity and communication efficiency, and FL for edge computing.

Criteria. Following (Mills, Hu, and Min 2022), we adopt the average User model Accuracy (UA) as the primary metric for evaluating model precision, focusing on the highest value achieved within 100 communication rounds. In addition, we assess communication efficiency by monitoring the learning curves, measuring average UA against per unit of communication overhead.

Performance Evaluation

Average User Model Accuracy. Table 1 displays the comparison of average UA on image recognition datasets, CIFAR10,

	Method	CIFAR-10		CINIC-10		CIFAR-100	
		$\alpha=0.5$	$\alpha=2$	$\alpha=0.5$	$\alpha=2$	$\alpha=0.5$	$\alpha=2$
Model Homo.	MTFL	31.1	29.2	32.1	34.8	14.8	15.4
	kNN-Per	32.7	34.8	32.8	29.6	18.8	18.3
	SCDPFL	49.4	33.1	48.7	32.2	33.3	19.6
	FedKD	40.9	23.9	39.2	22.7	26.1	14.3
	FedCache	42.1	23.9	39.8	21.9	26.4	14.7
	DistilCacheFL	**51.1**	**36.5**	**51.1**	**36.3**	**35.8**	**23.3**
	Method	**CIFAR-10**		**CINIC-10**		**CIFAR-100**	
		$\alpha=0.5$	$\alpha=2$	$\alpha=0.5$	$\alpha=2$	$\alpha=0.5$	$\alpha=2$
Model Hetero.	FedKD	39.7	24.1	39.6	23.6	26.2	14.1
	FedCache	41.3	22.2	40.3	22.4	26.3	13.9
	DistilCacheFL	**51.1**	**35.7**	**51.2**	**36.9**	**35.8**	**23.5**

Table 1: Average UA on image recognition tasks with two degrees of data heterogeneity.

Method	$\tau=0$	$\tau=0.3$	$\tau=0.5$	$\tau=0.7$	$\tau=1.0$
FedCache	41.2				
DistilCacheFL	51.3	**51.7**	51.1	49.8	48.6

Table 2: Ablation study on cache sampling strategy. Results are derived from the CIRAR-10 dataset with homogeneous models, taking $\alpha = 0.5$.

Method	Model	CIFAR10	CINIC10	CIFAR100
FedCache	ResNet-S	40.5	38.3	25.2
	ResNet-S/M/L	41.3	40.3	26.3
DistilCacheFL	ResNet-S	46.6	46.9	31.5
	ResNet-S/M/L	**51.1**	**51.1**	**35.8**

Table 3: Ablation study on model settings. Results are derived from image recognition tasks, taking $\alpha = 0.5$.

CIFAR100, and CINIC10, with two degrees of data heterogeneity, $\alpha \in \{0.5, 2.0\}$. As displayed, DistilCacheFL significantly outperforms all considered state-of-the-art methods across both model homogeneous and heterogeneous settings, demonstrating its superior performance and robustness in diverse edge scenarios. This substantial improvement is attributed to the enriched information characterization provided by distilled data and effective personalized optimization facilitated by device-centric cache sampling.

Communication Cost. Figure 4 illustrates the learning curves for image recognition tasks, plotting average UA against communication cost. As shown, DistilCacheFL exhibits significantly steeper convergence curves, reaching acceptable average UA more efficiently than competing methods, regardless of data heterogeneity, model structures, and datasets. This indicates that DistilCacheFL can achieve robust performance improvement with reduced communication overhead, making it suitable for deployment in edge environments with limited wireless bandwidth. The reduction in communication cost is attributed to DistilCacheFL's elimination of transferring cumbersome model parameters between devices and the server. Alternatively, DistilCacheFL leverages compact distilled data as knowledge to facilitate communication-efficient personalized optimization on devices.

Ablation Study

Impact of Cache Sampling Strategy. Table 2 presents the average UA with different τ values. We can conclude that increasing τ in the early stage generally improves the average UA due to the richer information provided by a greater number of cached samples. However, the performance gain diminishes as τ approaches 1. This decline is likely due to the introduction of data distribution bias across devices, which is harmful to the system's performance.

Impact of Model Settings. Table 3 presents the average UA for different model configurations across image recognition datasets. Results indicate that heterogeneous model settings yield higher average UA compared to homogeneous settings constrained by the weakest end devices. The improvements stem from the support of more powerful devices to deploy larger and more complex models, which can make full use of computational resources among heterogeneous devices to achieve better performance. These findings underscore the benefits of model heterogeneity flexibility in DistilCacheFL.

Discussion

Broader Impacts. Broader Impacts. DistilCacheFL enables flexible FEL participation under uncertain connectivity, eliminating the need for simultaneous device availability. This benefits dynamic networks and IoT applications vulnerable to power outages or poor signal. Its generalized data anonymization supports diverse data types and tasks, allowing seamless integration into smart healthcare and e-commerce systems. This facilitates deploying personalized models on edge devices (e.g., smartwatches, phones) for health monitoring and preference tracking.

Limitations. In terms of potential limitations of DistilCacheFL, devices may maliciously upload misleading or poisoned distilled data to the server, which could negatively affect the overall system performance. In addition, the dataset distillation process conducted on devices demands considerable computational resources. This request can somewhat lead to slower training procedures on devices with low hardware capabilities or those constrained by battery life.

Conclusion

In this paper, we introduce DistilCacheFL, a novel personalized FEL architecture to address the challenges of resource heterogeneity, communication limitations, and dynamic network conditions in demanding edge environments. By incorporating the benefits of both knowledge cache-driven federated learning and dataset distillation, DistilCacheFL facilitates privacy-preserving and semantically enriched knowledge organization and transfer among devices and the server. This is achieved through an iterative process of distilling data on devices, caching them on the server, and then dispatching the cached knowledge to guide local training. Moreover, we propose a device-centric cache sampling strategy to further enhance personalized model training by adapting to client data distributions and communication constraints. Extensive experiments on various tasks and datasets demonstrate that DistilCacheFL outperforms state-of-the-art methods with reduced communication costs.

Acknowledgments

This work was supported by the National Key Research and Development Program of China under Grant No. 2023YFB2703700 and the National Natural Science Foundation of China under Grant No. 62472410, and Institute of Computing Technology, Chinese Academy of Sciences - China Mobile Research Institute Joint Innovation Platform. Jingyuan Wang's work was partially supported by the National Natural Science Foundation of China (No. 72171013, 72222022, 72242101) and the Fundamental Research Funds for the Central Universities (JKF-2025017226182).

References

Al-Quraan, M.; Mohjazi, L.; Bariah, L.; Centeno, A.; Zoha, A.; Arshad, K.; Assaleh, K.; Muhaidat, S.; Debbah, M.; and Imran, M. A. 2023. Edge-native intelligence for 6G communications driven by federated learning: A survey of trends and challenges. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 7(3): 957–979.

Alam, S.; Liu, L.; Yan, M.; and Zhang, M. 2022. Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction. *Advances in neural information processing systems*, 35: 29677–29690.

Carpineti, C.; Lomonaco, V.; Bedogni, L.; Di Felice, M.; and Bononi, L. 2018. Custom dual transportation mode detection by smartphone devices exploiting sensor diversity. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 367–372. IEEE.

Chen, Z.; Yang, H.; Quek, T.; and Chong, K. F. E. 2023a. Spectral Co-Distillation for Personalized Federated Learning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 8757–8773. Curran Associates, Inc.

Chen, Z.; Yang, H.; Quek, T.; and Chong, K. F. E. 2023b. Spectral Co-Distillation for Personalized Federated Learning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 8757–8773. Curran Associates, Inc.

Cho, Y. J.; Manoel, A.; Joshi, G.; Sim, R.; and Dimitriadis, D. 2022. Heterogeneous Ensemble Knowledge Transfer for Training Large Models in Federated Learning. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2881–2887. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Darlow, L. N.; Crowley, E. J.; Antoniou, A.; and Storkey, A. J. 2018. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*.

Deng, Y.; Ren, J.; Tang, C.; Lyu, F.; Liu, Y.; and Zhang, Y. 2023. A hierarchical knowledge transfer framework for heterogeneous federated learning. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*, 1–10. IEEE.

Duan, Q.; Huang, J.; Hu, S.; Deng, R.; Lu, Z.; and Yu, S. 2023. Combining federated learning and edge computing toward ubiquitous intelligence in 6G network: Challenges, recent advances, and future directions. *IEEE Communications Surveys & Tutorials*.

Guo, Y.; Liu, F.; Cai, Z.; Zeng, H.; Chen, L.; Zhou, T.; and Xiao, N. 2021. PREFER: Point-of-interest REcommendation with efficiency and privacy-preservation via Federated Edge leaRning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(1): 1–25.

He, C.; Annavaram, M.; and Avestimehr, S. 2020. Group knowledge transfer: Federated learning of large cnns at the edge. *Advances in Neural Information Processing Systems*, 33: 14068–14080.

He, C.; Li, S.; So, J.; Zeng, X.; Zhang, M.; Wang, H.; Wang, X.; Vepakomma, P.; Singh, A.; Qiu, H.; et al. 2020. Fedml: A research library and benchmark for federated machine learning. *arXiv preprint arXiv:2007.13518*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Huang, W.; Ye, M.; and Du, B. 2022. Learn from others and be yourself in heterogeneous federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10143–10153.

Itahara, S.; Nishio, T.; Koda, Y.; Morikura, M.; and Yamamoto, K. 2023. Distillation-Based Semi-Supervised Federated Learning for Communication-Efficient Collaborative Training With Non-IID Private Data. *IEEE Transactions on Mobile Computing*, 22(1): 191–205.

Jin, H.; Bai, D.; Yao, D.; Dai, Y.; Gu, L.; Yu, C.; and Sun, L. 2022. Personalized edge intelligence via federated self-knowledge distillation. *IEEE Transactions on Parallel and Distributed Systems*, 34(2): 567–580.

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

Lee, G.; Jeong, M.; Shin, Y.; Bae, S.; and Yun, S.-Y. 2022. Preservation of the global knowledge by not-true distillation in federated learning. *Advances in Neural Information Processing Systems*, 35: 38461–38474.

Lei, S.; and Tao, D. 2023. A comprehensive survey of dataset distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Li, D.; and Wang, J. 2019. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*.

Li, T.; Sahu, A. K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; and Smith, V. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2: 429–450.

Liu, J.; Xu, H.; Wang, L.; Xu, Y.; Qian, C.; Huang, J.; and Huang, H. 2023. Adaptive Asynchronous Federated Learning in Resource-Constrained Edge Computing. *IEEE Transactions on Mobile Computing*, 22(2): 674–690.

Liu, L.; Zhang, J.; Song, S.; and Letaief, K. B. 2020. Client-edge-cloud hierarchical federated learning. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*, 1–6. IEEE.

Marfoq, O.; Neglia, G.; Vidal, R.; and Kameni, L. 2022. Personalized federated learning through local memorization. In *International Conference on Machine Learning*, 15070–15092. PMLR.

McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; and y Arcas, B. A. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, 1273–1282. PMLR.

Mills, J.; Hu, J.; and Min, G. 2022. Multi-Task Federated Learning for Personalised Deep Neural Networks in Edge Computing. *IEEE Transactions on Parallel and Distributed Systems*, 33(3): 630–641.

Nguyen, D. C.; Pham, Q.-V.; Pathirana, P. N.; Ding, M.; Seneviratne, A.; Lin, Z.; Dobre, O.; and Hwang, W.-J. 2022. Federated learning for smart healthcare: A survey. *ACM Computing Surveys (Csur)*, 55(3): 1–37.

Oh, S.; Park, J.; Jeong, E.; Kim, H.; Bennis, M.; and Kim, S.-L. 2020. Mix2FLD: Downlink federated learning after uplink federated distillation with two-way mixup. *IEEE Communications Letters*, 24(10): 2211–2215.

Reddi, S. J.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečný, J.; Kumar, S.; and McMahan, H. B. 2021. Adaptive Federated Optimization. In *International Conference on Learning Representations*.

Salamon, J.; Jacoby, C.; and Bello, J. P. 2014. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, 1041–1044.

Sattler, F.; Marban, A.; Rischke, R.; and Samek, W. 2021. Cfd: Communication-efficient federated distillation via soft-label quantization and delta coding. *IEEE Transactions on Network Science and Engineering*, 9(4): 2025–2038.

Shi, M.; Zhou, Y.; Wang, K.; Zhang, H.; Huang, S.; Ye, Q.; and Lv, J. 2023. PRIOR: Personalized Prior for Reactivating the Information Overlooked in Federated Learning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 28378–28392. Curran Associates, Inc.

Song, R.; Liu, D.; Chen, D. Z.; Festag, A.; Trinitis, C.; Schulz, M.; and Knoll, A. 2023. Federated learning via decentralized dataset distillation in resource-constrained edge environments. In *2023 International Joint Conference on Neural Networks (IJCNN)*, 1–10. IEEE.

Tak, A.; and Cherkaoui, S. 2021. Federated Edge Learning: Design Issues and Challenges. *IEEE Network*, 35(2): 252–258.

Tan, A. Z.; Yu, H.; Cui, L.; and Yang, Q. 2022. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*.

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

Wang, J.; Lin, Y.; and Li, Y. 2025. GTG: Generalizable Trajectory Generation Model for Urban Mobility. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 834–842.

Wang, L.; and Yoon, K.-J. 2021. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Wang, Z.; Xu, H.; Liu, J.; Huang, H.; Qiao, C.; and Zhao, Y. 2021. Resource-efficient federated learning with hierarchical aggregation in edge computing. In *IEEE INFOCOM 2021-IEEE conference on computer communications*, 1–10. IEEE.

Wang, Z.; Xu, H.; Liu, J.; Xu, Y.; Huang, H.; and Zhao, Y. 2022. Accelerating federated learning with cluster construction and hierarchical aggregation. *IEEE Transactions on Mobile Computing*.

Wen, T.; Zhang, H.; Zhang, H.; Wu, H.; Wang, D.; Liu, X.; Zhang, W.; Wang, Y.; and Cao, S. 2024. RTIFed: A Reputation based Triple-step Incentive mechanism for energy-aware Federated learning over battery-constricted devices. *Computer Networks*, 241: 110192.

Wu, C.; Wu, F.; Lyu, L.; Huang, Y.; and Xie, X. 2022. Communication-efficient federated learning via knowledge distillation. *Nature communications*, 13(1): 2032.

Wu, Z.; He, T.; Sun, S.; Wang, Y.; Liu, M.; Gao, B.; and Jiang, X. 2024a. Federated Class-Incremental Learning with New-Class Augmented Self-Distillation. *arXiv preprint arXiv:2401.00622*.

Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Gao, B.; Pan, Q.; He, T.; and Jiang, X. 2023a. Agglomerative federated learning: Empowering larger model training via end-edge-cloud collaboration. *arXiv preprint arXiv:2312.11489*.

Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Pan, Q.; Jiang, X.; and Gao, B. 2023b. FedICT: Federated Multi-task Distillation for Multi-access Edge Computing. *IEEE Transactions on Parallel and Distributed Systems*.

Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Pan, Q.; Zhang, J.; Li, Z.; and Liu, Q. 2024b. Exploring the distributed knowledge congruence in proxy-data-free federated distillation. *ACM Transactions on Intelligent Systems and Technology*, 15(2): 1–34.

Wu, Z.; Sun, S.; Wang, Y.; Liu, M.; Xu, K.; Wang, W.; Jiang, X.; Gao, B.; and Lu, J. 2024c. FedCache: A Knowledge Cache-Driven Federated Learning Architecture for Personalized Edge Intelligence. *IEEE Transactions on Mobile Computing*, 1–15.

Xue, J.; Sun, S.; Liu, M.; Wang, Y.; Liu, Z.; and Wang, J. 2025a. Learnable Sparse Customization in Heterogeneous Edge Computing. In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*, 58–71. IEEE Computer Society.

Xue, J.; Sun, S.; Liu, M.; Wang, Y.; Meng, X.; Wang, J.; Zhang, J.; and Xu, K. 2025b. Burst-Sensitive Traffic Forecast Via Multi-Property Personalized Fusion in Federated Learning. *IEEE Transactions on Mobile Computing*.

Yang, Q.; Liu, Y.; Chen, T.; and Tong, Y. 2019. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2): 1–19.

Yang, Z.; Zhang, Y.; Zheng, Y.; Tian, X.; Peng, H.; Liu, T.; and Han, B. 2023. FedFed: Feature Distillation against Data Heterogeneity in Federated Learning. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 60397–60428. Curran Associates, Inc.

Yu, R.; and Li, P. 2021. Toward resource-efficient federated learning in mobile edge computing. *IEEE Network*, 35(1): 148–155.

Yuan, W.; Yin, H.; Wu, F.; Zhang, S.; He, T.; and Wang, H. 2023. Federated unlearning for on-device recommendation. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 393–401.

Zhang, T.; Gao, L.; He, C.; Zhang, M.; Krishnamachari, B.; and Avestimehr, A. S. 2022. Federated learning for the internet of things: Applications, challenges, and opportunities. *IEEE Internet of Things Magazine*, 5(1): 24–29.

Zhu, Z.; Hong, J.; Drew, S.; and Zhou, J. 2022. Resilient and communication efficient learning for heterogeneous federated systems. *Proceedings of machine learning research*, 162: 27504.

Zhu, Z.; Hong, J.; and Zhou, J. 2021. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, 12878–12889. PMLR.