MetaFed: Federated Learning among Federations with Cyclic Knowledge Distillation for Personalized Healthcare

Yiqiang Chen[1,2,3], **Wang Lu**[1,2], **Xin Qin**[1,2], **Jindong Wang**[4], **Xing Xie**[4]
[1]Beijing Key Lab. of Mobile Computing and Pervasive Devices, Inst. of Comp. Tech., CAS
[2]University of Chinese Academy of Sciences, Beijing, China
[3]Peng Cheng Laboratory, Shenzhen, China [4]Microsoft Research Asia, Beijing, China
{yqchen,luwang,qinxin18b}@ict.ac.cn, {jindong.wang, xingx}@microsoft.com

Abstract

Federated learning has attracted increasing attention to building models without accessing the raw user data, especially in healthcare. In real applications, different federations can seldom work together due to possible reasons such as data heterogeneity and distrust/inexistence of the central server. In this paper, we propose a novel framework called **MetaFed** to facilitate trustworthy FL between different federations. MetaFed obtains a personalized model for each federation without a central server via the proposed Cyclic Knowledge Distillation. Specifically, MetaFed treats each federation as a meta distribution and aggregates knowledge of each federation in a *cyclic* manner. The training is split into two parts: common knowledge accumulation and personalization. Comprehensive experiments on three benchmarks demonstrate that MetaFed without a server achieves better accuracy compared to state-of-the-art methods (e.g., 10%+ accuracy improvement compared to the baseline for PAMAP2) with fewer communication costs.

1 Introduction

Machine learning, especially deep learning, has become prominent in people's daily lives [Lu *et al.*, 2022a; Lu *et al.*, 2022b; He *et al.*, 2021]. It is applied to many health-related fields such as human activity recognition [Lu *et al.*, 2021], medical images [Li *et al.*, 2021a], and other fields [Ma *et al.*, 2021]. However, with the increasing awareness of data privacy and security, some countries and organizations released policies to prevent data leakage [Inkster, 2018; Voigt and Von dem Bussche, 2017]. In this situation, federated learning (FL) [Yang *et al.*, 2019] was proposed and has attracted increasing attention recently.

Google [McMahan *et al.*, 2017] proposed the first FL algorithm called FedAvg to aggregate clients' information. FedAvg replaces direct data exchanges with model parameter communication to preserve data privacy. Although FedAvg achieves promising performance in many applications, it may not be feasible in more challenging trustworthy FL situations, e.g. medical institutions may be grouped into multiple federations and no higher level governing organizations exist. As shown in Figure 1, a certain number of clients form a federation, and different federations are independent enough that do not use a central server, but communicate with each other instead. Inside each federation, different FL algorithms can be used to train a model. However, it remains unclear how to build personalized FL models outside the federations, i.e., FL among different federations. Moreover, data heterogeneity widely exists in these federations. We view each federation as a meta distribution and view the problem in this situation as meta federated learning.[1]

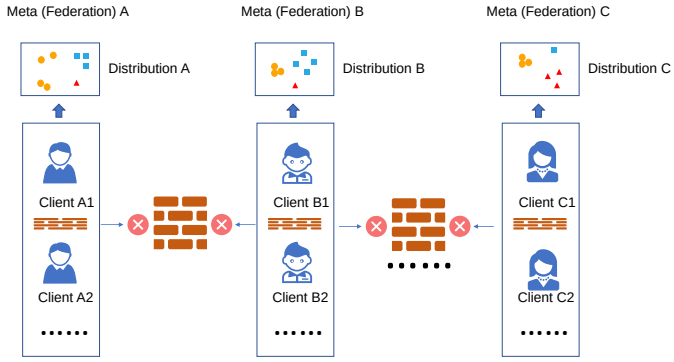


Figure 1: Issues in meta federated learning.

In this article, we propose **MetaFed**, a meta federated learning framework for cross-federation federated learning. We focus on inter-federation federated learning in this paper and each federation can be viewed as an independent individual. To implement MetaFed, we propose a cyclic knowledge distillation method. MetaFed can solve data islanding and data statistical heterogeneity without requiring a server or sacrificing user privacy. Specifically, MetaFed consists of two stages, common knowledge accumulation stage and personalization stage. In the first stage, the model trained on the previous meta federation serves as the teacher for the next federation and knowledge distillation (KD) [Hinton *et al.*, 2015; Romero *et al.*, 2014] aims to make use of the common information. Several rounds with fixed hyperparameters for knowledge distillation are performed to ensure enough common knowledge. In personalization stage, we utilize KD with adapted hyperparameters to obtain personalized model for

[1]We use meta and federation interchangeably.

each federation. Through knowledge distillation, it can not only acquire common knowledge among federations but also cope with feature shifts and label shifts. Moreover, MetaFed is extensible and can be deployed to many applications. The code for MetaFed is released at https://github.com/microsoft/PersonalizedFL.

Our contributions are as follows.

1. We propose MetaFed, a novel meta federated learning framework via cyclic knowledge distillation for healthcare, which can accumulate common information from different federations without compromising privacy security, and achieve personalized models for each federation through adapted knowledge distillation.
2. Comprehensive experiments on image and time-series datasets illustrate that MetaFed has remarkable performance in each federation without a server compared to state-of-the-art methods. Moreover, MetaFed reduces the number of rounds, thus saving communication costs.
3. MetaFed is extensible and can be applied in many healthcare applications, which means it can work well in many circumstances. We can even replace the knowledge distillation with some other incremental learning methods for specific applications.

2 Related Work

2.1 Federated Learning

To make full use of data in different separate clients and protect data privacy and security simultaneously, Google [2017] first proposed FedAvg to train machine learning models via aggregating distributed mobile phones' data with exchanging model parameters instead of directly exchanging data. FedAvg can work well with data islanding problems in many applications although it is simple. Subsequently, Yang et al. [2019] wrote the first survey of FL research.

Federated learning has attracted growing attention in many applications. And the traditional and simple FedAvg cannot satisfy complicated realistic scenes. When meeting data statistical heterogeneity, FedAvg may converge slowly and acquire large amounts of communication cost. Moreover, since only a shared global model is obtained, the model may degrade when predicting in personalized clients. Some work tries to cope with these problems. FedProx [Li *et al.*, 2018] added a proximal term to FedAvg which referred to the global model and allowed slight differences when training local models. Yu et al. [2020] combined three traditional adaptation techniques: fine-tuning, multi-task learning, and knowledge distillation into federated models. Most recently, FedBN [Li *et al.*, 2021b] tried to cope with feature shifts among clients via preserving local batch normalization parameters which can represent data distributions to some extent. Some other work made an effort to utilize personalization federated learning in the healthcare field [Chen *et al.*, 2020; Lu *et al.*, 2022c]. [Chen *et al.*, 2020] proposed a federated transfer learning framework which needs some sharing data while [Lu *et al.*, 2022c] proposed FedAP which could achieve remark personalized performance with few rounds via aggregating with clients' similarities. However, these methods need a server and have some limits in communication costs.

In this situation where no server exists, FedAvg even cannot be implemented. Sequential training may be a reasonable solution. [Kopparapu and Lin, 2020] proposed FedFMC that dynamically forked devices into updating different global models and merged models in a lifelong way. [Zaccone *et al.*, 2022] leveraged the sequential training of subgroups of heterogeneous clients to emulate the centralized paradigm. [Zeng *et al.*, 2022] assigned clients to homogeneous groups to minimize the overall distribution divergence among groups. These methods still rely heavily on parallel federated learning where sequential training round style with only one round and no closed loop is just an aid.

Some other work, e.g. [Roy *et al.*, 2019; Rieke *et al.*, 2020; Li *et al.*, 2021c], communicated in a peer-to-peer environment without a server. BrainTorrent [Roy *et al.*, 2019] presented a highly dynamic peer-to-peer environment, where all centers directly interacted with each other without depending on a central body. It seemed disorderly and chaotic, and it required lots of communication costs. Nicola Rieke et al. [Rieke *et al.*, 2020] considered key factors to federated learning while FedH2L [Li *et al.*, 2021c] utilized mutual distillation to exchange posteriors on a shared seed set between participants in a decentralized manner. However, these methods often require large communication costs, and few are designed for personalization federation federated learning. No work pays attention to proposing a new paradigm for personalized federated learning among federations without a server.

2.2 Knowledge Distillation

Knowledge distillation has been a well-known technique to transfer knowledge since birth [Hinton *et al.*, 2015]. In the original version, the knowledge was transferred by mimicking the outputs of the teacher model on the same data. Later, besides imitating outputs, some work demonstrated that feature imitation could also guide the student model training [Romero *et al.*, 2014]. Nowadays, as a common technique, knowledge distillation is also applied to federated learning [Usmanova *et al.*, 2021; Afonin and Karimireddy, 2021]. Though mimicking the global model and the local previous model, different implementations can be flexibly applied to different situations.

3 Method

3.1 Problem Formulation

In a personalized federated learning among federations setting, N different federations, denoted as $\{F_1, F_2, \cdots, F_N\}$, have data, denoted as $\{\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_N\}$ with different distributions, which means $P(\mathcal{D}_i) \neq P(\mathcal{D}_j)$. For simplicity, we only study the case where the input and output spaces are the same, i.e. $\mathcal{X}_i = \mathcal{X}_j, \mathcal{Y}_i = \mathcal{Y}_j, \forall i \neq j$. Each dataset, $\mathcal{D}_i = \{(\mathbf{x}_{i,j}, y_{i,j})\}_{j=1}^{n_i}$, consists of three parts, a train dataset $\mathcal{D}_i^{train} = \{(\mathbf{x}_{i,j}^{train}, y_{i,j}^{train})\}_{j=1}^{n_i^{train}}$, a validation dataset $\mathcal{D}_i^{valid} = \{(\mathbf{x}_{i,j}^{valid}, y_{i,j}^{valid})\}_{j=1}^{n_i^{valid}}$ and a test dataset $\mathcal{D}_i^{test} = \{(\mathbf{x}_{i,j}^{test}, y_{i,j}^{test})\}_{j=1}^{n_i^{test}}$. We have $n_i = n_i^{train} +$

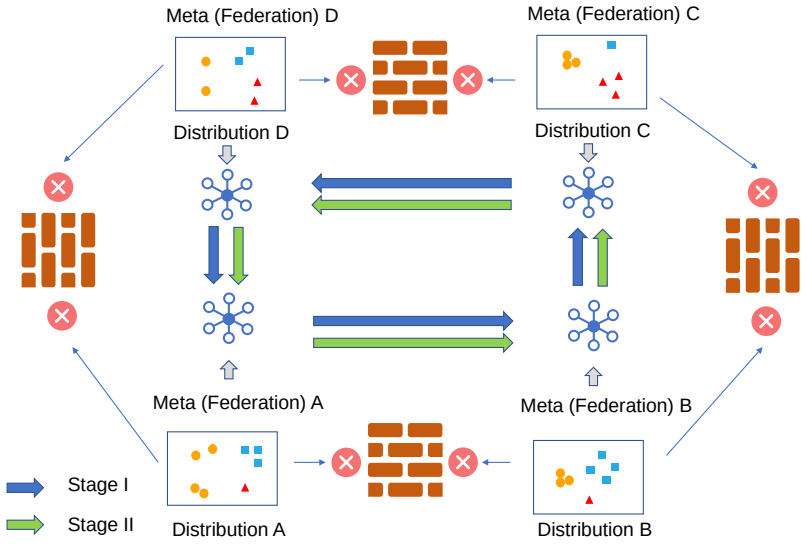


Figure 2: The structure of MetaFed for federated learning among federations. Stage I is the common knowledge accumulation stage where the model is sent after local training. Stage II is the personalization stage where the common knowledge model is sent before local training.

$n_i^{valid} + n_i^{test}$ and $\mathcal{D}_i = \mathcal{D}_i^{train} \cup \mathcal{D}_i^{valid} \cup \mathcal{D}_i^{test}$. We aim to combine information of all federations without data exchange to learn a good model f_i for each federation on its local dataset $\mathcal{D}_i$:

$$\min_{\{f_k\}_{k=1}^N} \frac{1}{N} \sum_{i=1}^{N} \frac{1}{n_i^{test}} \sum_{j=1}^{n_i^{test}} \ell(f_i(\mathbf{x}_{i,j}^{test}), y_{i,j}^{test}), \qquad (1)$$

where ℓ is a loss function.

3.2 Overview of MetaFed

Consider the union of different federations where there is no server among them and distribution shifts exist. How to make them communicate equally without any other governors and share common knowledge without direct data exchange is the key. MetaFed aims to accumulate common knowledge and preserve personalized information without compromising data privacy and security via knowledge distillation in a cyclic way. Figure 2 gives an overview.

Without loss of generality, we assume there are four federations, and it can be extended to the more general case easily. As shown in Figure 2, the whole training process is split into two stages, common knowledge accumulation stage (blue arrows) and personalization stage (green arrows). In common knowledge accumulation stage, the federations are trained in order and the previous trained one serves as the teacher for the next one. The common knowledge accumulation stage lasts for several rounds to ensure each federation's common knowledge are extracted completely. Personalization stage is also trained in the same style but the model is sent to the next federation without local training for losing no common knowledge. From Figure 2, we can see no server participates in the training process. The two stages are both based on feature knowledge distillation (as shown in Figure 3),

$$\ell_{dist}(g_{tea}, g_{stu}; \mathbf{x}) = ||g_{tea}(\mathbf{x}) - g_{stu}(\mathbf{x})||_2^2, \qquad (2)$$

where g_{tea} is the feature extractor of the previous federation while g_{stu} is for the current training federation, and $\mathbf{x}$ is a sample of data from the current federation. Through knowledge distillation, we can make good use of knowledge, viewed as common knowledge, from the previous federation. Therefore, the total loss to train the local model, f_i, is,

$$\ell_{total}^i = \frac{1}{n_i^{train}} \sum_{(\mathbf{x},y)\in\mathcal{D}_i^{train}} \ell_{cls}(f_i; \mathbf{x}, y) + \lambda \ell_{dist}(g_{tea}, g_i; \mathbf{x}), \qquad (3)$$

where λ is a trade-off of knowledge transfer and focusing the current data while ℓ_{cls} is the cross-entropy loss. $f_i = c_i \circ g_i$ where c_i is the classification layer and g_i is the feature extractor. In the following, we will specify the two stages respectively and illustrate how to design λ in detail.

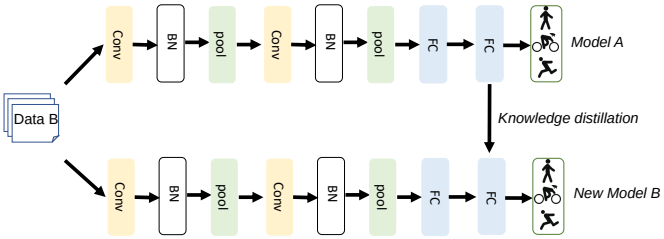


Figure 3: The knowledge transfer between two federations.

3.3 Common Knowledge Accumulation Stage

This stage happens in the first part of the whole training process. In this stage, federations are trained sequentially in a cyclic way and the previous meta knowledge is transferred to the next one via knowledge distillation. The knowledge which is useful for the current federation will be preserved via knowledge distillation while the others that are useless will be discarded. With several rounds of cyclic training, the knowledge that is useful for all federations will be preserved

and we denote this type of knowledge as common knowledge. Our first stage is just to accumulate common knowledge and the specific detail can be found in Algorithm 1.

Algorithm 1 Common knowledge accumulation stage

Input: N federations' datasets $\{\mathcal{D}_i\}_{i=1}^{N}$, λ_0, l_{t1}
Output: A common model f

Initial $\lambda = \lambda_0$
Train local models, f_i using ℓ_{cls} with $\mathcal{D}_i^{train}$ in each federation
Send the current model f_i to the next federation $i+1$
Evaluate f_{i+1} on $\mathcal{D}_{i+1}^{valid}$ and obtain acc_{i+1}^{valid}
if $acc_{i+1}^{valid} > l_{t1}$ **then**
    Train f_{i+1} with Eq. (3)
else
    Use f_i to initial f_{i+1}
    Train f_{i+1} using ℓ_{cls} with $\mathcal{D}_{i+1}$
end if
Repeat steps $3 \sim 10$ until convergence
The last model f_N serve as the final common model, f

As shown in Algorithm 1, the valid accuracy on the current federation's valid data is used to determine whether to complete preserve the previous federation's knowledge. When $acc_{i+1}^{valid} > l_{t1}$, we think the training data on the current federation have enough knowledge to train a model, which means we can discard some unless knowledge via knowledge distillation. When $acc_{i+1}^{valid} < l_{t1}$, the training data on the current federation has too little information. We need to make full use of the previous knowledge and thereby we directly initial the current model with the previous one. To preserve the personalization, we may preserve the local BN following FedBN [Li *et al.*, 2021b]. Since we want to accumulate common knowledge in this stage, we fix $\lambda = \lambda_0$ to ensure enough common knowledge preserved.

3.4 Personalization Stage

This stage happens in the second part of the whole training process. In the above stage, we obtain the common model f which contains enough common knowledge. Since no server exists, we have to obtain the personalization models in the same style (sequential) as the first stage. To prevent common knowledge loss, we transmit the common f to the next federation before local training. The specific detail of the second stage can be found in Algorithm 2.

When the common model performs seriously terribly on the validation data of the current federation, we want to refer little on it and thereby set $\lambda = 0$. In the first stage, the current f_i has contained other federations' knowledge. When the common model's performance is acceptable on the current validation data, we adapt λ for personalization:

$$\lambda = \lambda_0 \times 10^{\min\left(1,(acc_{i,i+1}^{valid} - acc_{i+1,i+1}^{valid})*5\right)-1}. \quad (4)$$

Compared to the local model's performance, the better the common model's performance is, the larger λ will be.

Algorithm 2 Personalization stage

Input: N federations' datasets $\{\mathcal{D}_i\}_{i=1}^{N}$, λ_0, l_{t2}, f
Output: Meta models $\{f_i\}_{i=1}^{N}$

Send the common model f to the next federation $i+1$
Evaluate f_i on $\mathcal{D}_{i+1}^{valid}$ and obtain $acc_{i,i+1}^{valid}$
Evaluate f_{i+1} on $\mathcal{D}_{i+1}^{valid}$ and obtain $acc_{i+1,i+1}^{valid}$
if $acc_{i,i+1}^{valid} \leq acc_{i+1,i+1}^{valid}$ and $acc_{i,i+1} < l_{t2}$ **then**
    Set $\lambda = 0$
else
    Set λ via Eq. (4)
end if
Train f_{i+1} with Eq. (3)
Repeat steps $1 \sim 9$ until all f_i are trained

4 Experiments

We evaluate the performance of MetaFed on three benchmarks, including time series and image modalities. One is a famous benchmark (VLCS) about feature shifts while the others are both healthcare related.

We compare our method with three state-of-the-art methods including common federated learning methods and some federated learning methods designed for non-iid data,

- FedAvg [McMahan *et al.*, 2017]. Directly aggregate models' parameters without personalization.
- FedProx [Li *et al.*, 2018]. Allow slight differences between the local model and the global model via a proximal term added to FedAvg.
- FedBN [Li *et al.*, 2021b]. Preserve the local batch normalization not affected by the other clients.

Since these methods all need a server, we ease this restriction for them. Adapting these methods without a server will increase communication costs with no performance improvement. All methods use the same model for fairness.

4.1 Image Classification on Feature Shifts

VLCS. First, we adopt a public image classification dataset called VLCS [Fang *et al.*, 2013] due to few famous feature shift datasets in the healthcare field. VLCS comprises four photographic sub-datasets (Caltech101, LabelMe, SUN09, VOC2007) with 10,729 instances of 5 classes. Each sub-dataset serves as one federation and there are 4 federations in total. Since each dataset contains too many images, we choose 10% for training, 10% for validation, and 20% for testing. The validation parts are utilized to guide training and select the best model for each federation.

Implementation Details. For VLCS, we adopt Alexnet [Krizhevsky *et al.*, 2012] as the feature extractor and a three-layer fully connected neural network as the classifier. For model training, we use SGD optimizer with a learning rate of 10^{-2}. All methods are implemented in the same environment for fairness and we run three trials to record the average accuracy.

method	Caltech101	LabelMe	SUN09	VOC2007	AVG
FedAvg	82.69	54.43	51.52	44.89	58.38
FedProx	83.04	55.74	51.98	47.70	59.62
FedBN	90.81	54.80	50.15	44.30	60.02
MetaFed	**93.64**	**57.44**	**56.40**	**48.15**	**63.91**

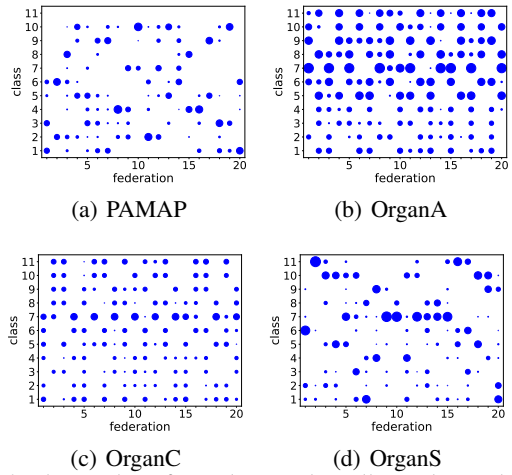
Table 1: Average accuracy on VLCS. Bold means the best result while underline means the second-best result.

Results. The classification results for each federation on VLCS are shown in Table 1. We have the following observations from these results. 1) Our method achieves the best effects on average with a remarkable improvement (over 3.89% compared to FedBN). We even achieve the best performance on each federation which demonstrates the superiority and the capability of personalization of our method. 2) Since it is a feature shift situation, FedBN has a better performance compared to FedAvg. FedProx has an acceptable performance.

4.2 Human Activity Recognition

PAMAP. For the healthcare-related benchmark, we first adopt a public time-series benchmark called PAMAP [Reiss and Stricker, 2012]. PAMAP contains data of 18 human activities performed by 9 subjects. We use 3 inertial measurement units' data with 27 channels and utilize the sliding window technique to preprocess data. 10 classes with 17639 instances are selected. To simulate labels shift, we follow [Yurochkin *et al.*, 2019] and use Dirichlet distributions to create disjoint non-iid. training data. Figure 4(a) visualized how samples are distributed. For each federation, $40\%, 30\%$, and 30% of data are used for training, validation, and testing respectively.

Implementation Details. For PAMAP, we utilize a CNN composed of two convolutional layers, two pooling layers, two batch normalization layers, and two fully connected layers [Wang *et al.*, 2019]. Other settings are similar to VLCS.



(a) PAMAP (b) OrganA

(c) OrganC (d) OrganS

Figure 4: The number of samples per class allocated to each federation (indicated by dot size).

Results. The classification results for each federation on PAMAP are shown in Table 2. We have the following observations from these results. 1) Our method also achieves the best effects on average with a remarkable improvement (over 3.09% compared to FedBN) in this situation where label shifts exist. 2) In this situation, 20 federations in total make the problem more complicated. Although our method does not achieve the best performance on each federation, it still achieves acceptable results on almost every federation. 3) Compared to FedAvg and FedProx, FedBN and our method achieve remarkable improvement, which may illustrate that FedBN can also cope with label shifts sometimes.

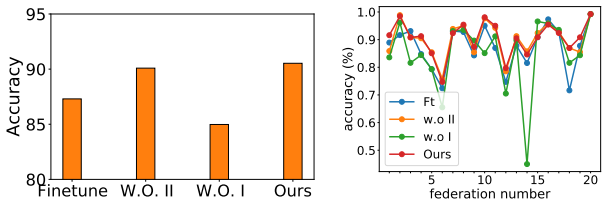
4.3 Real Medical Image Classification

MedMnist. To further validate our method, we evaluate our method on three public medical image classification benchmarks. We choose 3 datasets, OrganAMNIST, OrganCMNIST, and OrganSMNIST [Bilic *et al.*, 2019; Xu *et al.*, 2019], from a larger-scale MNIST-like collection of standardized biomedical images, MedMNIST [Yang *et al.*, 2021a; Yang *et al.*, 2021b]. These three datasets are all about Abdominal CT images with 11 classes and they have 58,850, 23,660, and 25,221 samples respectively. Similar to PAMAP, we utilize Dirichlet distribution to split data and Figure 4(b)-Figure 4(d) visualize how samples are distributed. In each federation, $40\%, 30\%$, and 30% data are used for training, validation and testing respectively.

Implementation Details. For these three datasets, we utilize adapted LeNet5 [LeCun *et al.*, 1998] due to the image size with 28×28. Other settings are similar to VLCS.

Results. The classification results for each federation on OrganAMNIST, OrganCMNIST, and OrganSMNIST, are shown in Table 3. We have the following observations from these results. 1) Our method also achieves the best effects on average with a remarkable improvement (over $3.07\%, 3.08\%, 12.44\%$ respectively) in this situation where label shifts exist. 2) When federation distributions have small differences (Figure 4(b) and Figure 4(c)), three state-of-the-art methods have similar performance and ours achieves remarkable improvements. When federations have huge differences from each other (Figure 4(d)), FedBN can achieve a remarkable improvement compared to FedAvg and FedProx while ours shows another crazy improvement compared to FedBN 3) The above experiments demonstrate that our method can achieve the best performance in both two settings.

4.4 Analysis



(a) Average Acc (b) Federation Acc

Figure 5: Ablation study on PAMAP.

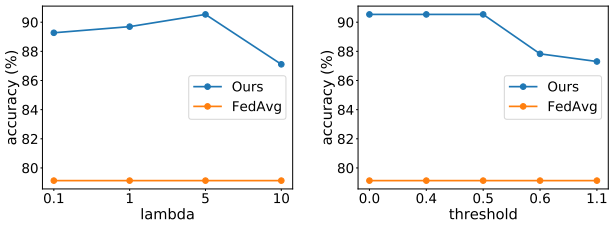
method	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	AVG
FedAvg	82.44	80.99	50.19	91.25	81.99	68.97	76.63	90.08	74.71	90.46	68.20	67.82	**92.02**	79.62	58.17	92.78	79.09	71.65	87.79	97.70	79.13
FedProx	82.44	81.37	52.49	92.02	82.38	70.50	77.78	90.84	78.16	89.31	69.35	70.88	90.87	79.62	63.88	93.92	77.95	71.26	85.88	96.93	79.89
FedBN	84.73	84.79	81.61	**93.54**	81.23	**80.08**	89.27	94.66	80.08	95.80	87.36	78.16	86.69	**88.08**	86.31	**96.58**	93.54	80.08	87.02	**99.23**	87.44
MetaFed	**91.60**	**98.48**	**90.80**	91.25	**85.06**	74.71	**92.34**	**95.42**	**87.36**	**98.09**	**95.02**	**79.69**	90.49	84.62	**90.87**	95.44	**92.40**	**86.97**	**90.84**	**99.23**	**90.53**

Table 2: Average accuracy on PAMAP. Bold means the best result while underline means the second-best result.

Bechmark	Method	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	AVG
OrganA	fedavg	94.20	91.23	92.61	89.85	96.25	88.62	90.44	94.65	92.84	89.29	91.81	92.94	89.62	93.84	93.86	89.65	93.52	90.43	92.26	92.38	92.01
	fedprox	93.75	91.34	92.95	90.76	96.47	89.65	90.79	94.65	93.30	89.29	91.81	91.69	90.99	93.50	92.83	89.65	92.83	90.43	92.04	91.92	92.03
	fedbn	94.09	92.03	93.40	90.31	97.27	88.96	92.49	95.90	94.89	89.18	92.15	91.69	91.11	92.25	92.72	90.33	93.52	91.00	92.04	91.13	92.32
	MetaFed	**96.59**	**93.39**	**97.27**	**94.30**	**97.84**	**90.79**	**94.31**	**96.47**	**97.16**	**94.42**	**96.47**	**94.87**	**93.84**	**96.47**	**97.16**	**94.20**	**96.47**	**94.65**	**95.79**	**95.34**	**95.39**
OrganC	fedavg	86.40	90.00	88.03	88.39	84.05	90.31	78.35	95.16	83.52	91.45	85.76	90.03	86.04	90.63	86.36	91.43	86.89	92.33	85.76	91.76	88.13
	fedprox	87.25	90.29	85.76	87.82	85.47	89.74	79.20	96.58	84.09	90.88	86.89	91.74	87.18	91.76	87.50	92.29	87.75	90.91	86.04	92.33	88.57
	fedbn	82.44	94.86	92.02	91.50	86.32	92.88	78.92	97.44	86.36	90.03	85.19	90.60	83.76	92.90	85.80	94.29	88.03	92.90	82.34	92.90	89.07
	MetaFed	**88.95**	**95.43**	**92.02**	**92.92**	**89.17**	90.60	**82.91**	**98.29**	**89.20**	**93.16**	**90.60**	**94.59**	**89.46**	**95.17**	**94.60**	**94.29**	**92.88**	**94.89**	**89.74**	**94.03**	**92.15**
OrganS	fedavg	52.39	50.26	67.47	67.82	75.67	53.33	82.98	61.97	82.35	90.72	65.78	83.42	73.53	90.13	83.11	46.42	44.80	61.17	56.38	79.41	68.46
	fedprox	52.13	51.59	69.07	67.82	77.01	53.33	81.38	59.84	82.89	90.19	59.36	86.10	71.12	89.33	83.65	45.89	43.20	62.77	58.78	79.68	68.26
	fedbn	75.00	76.98	75.20	79.79	79.95	54.13	86.97	67.29	92.51	99.20	75.67	86.90	76.47	90.93	89.28	72.15	57.87	69.95	73.67	87.43	78.37
	MetaFed	**97.07**	**97.62**	**81.07**	**85.37**	**86.36**	**89.87**	**97.34**	**98.14**	**95.19**	**99.73**	**85.83**	**88.24**	**83.96**	**97.07**	**90.08**	**81.96**	**86.13**	**85.37**	**96.28**	**93.58**	**90.81**

Table 3: Average accuracy on three benchmarks of MedMnist. Bold means the best result while underline means the second-best result.

Ablation Study. We also perform ablation study to illustrate the effects of each part of our methods. As shown in Figure 5(a) and Figure 5(b), we can see that both replacing knowledge distillation with fine-tuning (Finetune) and training without common knowledge accumulation stage (W.O. I) will make performance drop while common knowledge accumulation with slight adaptation (W.O. II) can achieve acceptable results. We do not use the common f for testing but each local model with local adaptation brought by knowledge distillation does testing. Personalization stage can further bring slightly better performance, which demonstrates that each part of our method can all bring benefits.



(a) λ_0 (b) l_{t1}

Figure 6: Parameter sensitivity on PAMAP.

Parameter Sensitivity. In this part, we evaluate the two main hyperparameters, λ_0 and l_{t1}, in MetaFed. We change one parameter and fix the others. Figure 6(a) proves that our method can achieve better performance than FedAvg whatever λ_0 is while Figure 6(b) demonstrates that our method may slightly drop with larger l_{t1} but it is still better than FedAvg. The results reveal that MetaFed is more effective and robust than other methods under different hyperparameters in most cases.

Communication Costs. To further prove that our method can reduce communication costs, we increase the local training iteration number and decrease the total communication rounds to evaluate our method and the baseline. As shown in Figure 7, when communication costs are limited, our method has related stable results while FedAvg drops seriously. In realistic applications, communication cost is an important evaluation metric and there are often strictly limited costs. Therefore, few communication costs with stable and acceptable performance are vital.

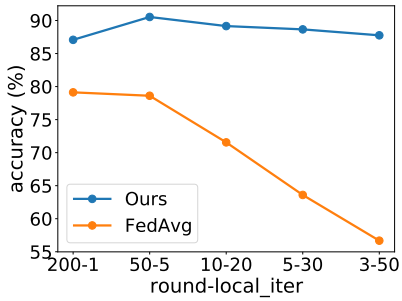


Figure 7: Performance w. communication costs on PAMAP.

5 Conclusion and Future Work

In this article, we proposed MetaFed which uses cyclic knowledge distillation for meta federated learning. MetaFed organizes federations in another novel style that does not require a central server. Comprehensive experiments have demonstrated the effectiveness of MetaFed. In the future, we plan to combine MetaFed with common methods such as FedAvg, to implement a complete federated learning system, including intra- and inter- federations. We also plan to extend MetaFed for heterogeneity architectures and apply MetaFed to more realistic healthcare applications.

ACKNOWLEDGMENTS

This work is supported by the National Key Research & Development Program of China No.2019YFB1404703, Natural Science Foundation of China (No.61972383, No.61902377, No.61902379), Science and Technology Service Network Initiative, Chinese Academy of Sciences (No.KFJ-STS-QYZD-2021-11-001).

References

[Afonin and Karimireddy, 2021] Andrei Afonin and Sai Praneeth Karimireddy. Towards model agnostic federated learning using knowledge distillation. *arXiv*, 2021.

[Bilic *et al.*, 2019] Patrick Bilic, Patrick Ferdinand Christ, Eugene Vorontsov, Grzegorz Chlebus, Hao Chen, Qi Dou, Chi-Wing Fu, Xiao Han, Pheng-Ann Heng, Jürgen Hesser, et al. The liver tumor segmentation benchmark (lits). *arXiv preprint arXiv:1901.04056*, 2019.

[Chen *et al.*, 2020] Yiqiang Chen, Xin Qin, Jindong Wang, Chaohui Yu, and Wen Gao. Fedhealth: A federated transfer learning framework for wearable healthcare. *IEEE Intelligent Systems*, 35(4):83–93, 2020.

[Fang *et al.*, 2013] Chen Fang, Ye Xu, and Daniel N Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1657–1664, 2013.

[He *et al.*, 2021] Mingjie He, Jie Zhang, Shiguang Shan, Xiao Liu, Zhongqin Wu, and Xilin Chen. Locality-aware channel-wise dropout for occluded face recognition. *IEEE Transactions on Image Processing*, 31:788–798, 2021.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7), 2015.

[Inkster, 2018] Nigel Inkster. *China's cyber power*. Routledge, 2018.

[Kopparapu and Lin, 2020] Kavya Kopparapu and Eric Lin. Fedfmc: Sequential efficient federated learning on non-iid data. *arXiv preprint arXiv:2006.10937*, 2020.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, volume 25, pages 1097–1105, 2012.

[LeCun *et al.*, 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[Li *et al.*, 2018] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

[Li *et al.*, 2021a] Shaohua Li, Xiuchao Sui, Xiangde Luo, Xinxing Xu, Yong Liu, and Rick Siow Mong Goh. Medical image segmentation using squeeze-and-expansion transformers. In *IJCAI*, 2021.

[Li *et al.*, 2021b] Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fedbn: Federated learning on non-iid features via local batch normalization. *arXiv preprint arXiv:2102.07623*, 2021.

[Li *et al.*, 2021c] Yiying Li, Wei Zhou, Huaimin Wang, Haibo Mi, and Timothy M Hospedales. Fedh2l: Federated learning with model and statistical heterogeneity. *arXiv*, 2021.

[Lu *et al.*, 2021] Wang Lu, Yiqiang Chen, Jindong Wang, and Xin Qin. Cross-domain activity recognition via substructural optimal transport. *Neurocomputing*, 454:65–75, 2021.

[Lu *et al.*, 2022a] Wang Lu, Jindong Wang, and Yiqiang Chen. Local and global alignments for generalizable sensor-based human activity recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

[Lu *et al.*, 2022b] Wang Lu, Jindong Wang, Yiqiang Chen, Sinno Pan, Chunyu Hu, and Xin Qin. Semantic-discriminative mixup for generalizable sensor-based cross-domain activity recognition. *IMWUT*, 2022.

[Lu *et al.*, 2022c] Wang Lu, Jindong Wang, Yiqiang Chen, Xin Qin, Renjun Xu, Dimitrios Dimitriadis, and Tao Qin. Personalized federated learning with adaptive batchnorm for healthcare. *IEEE Transactions on Big Data*, 2022.

[Ma *et al.*, 2021] Fenglong Ma, Muchao Ye, Junyu Luo, Cao Xiao, and Jimeng Sun. Advances in mining heterogeneous healthcare data. In Feida Zhu, Beng Chin Ooi, and Chunyan Miao, editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 4050–4051. ACM, 2021.

[McMahan *et al.*, 2017] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[Reiss and Stricker, 2012] Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers*, pages 108–109. IEEE, 2012.

[Rieke *et al.*, 2020] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

[Romero *et al.*, 2014] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[Roy *et al.*, 2019] Abhijit Guha Roy, Shayan Siddiqui, Sebastian Pölsterl, Nassir Navab, and Christian Wachinger. Braintorrent: A peer-to-peer environment for decentralized federated learning. *arXiv*, 2019.

[Usmanova *et al.*, 2021] Anastasiia Usmanova, François Portet, Philippe Lalanda, and German Vega. A distillation-based approach integrating continual learning and federated learning for pervasive services. *arXiv*, 2021.

[Voigt and Von dem Bussche, 2017] Paul Voigt and Axel Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 10:3152676, 2017.

[Wang *et al.*, 2019] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11, 2019.

[Xu *et al.*, 2019] Xuanang Xu, Fugen Zhou, Bo Liu, Dongshan Fu, and Xiangzhi Bai. Efficient multiple organ localization in ct image using 3d region proposal network. *IEEE transactions on medical imaging*, 38(8):1885–1898, 2019.

[Yang *et al.*, 2019] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[Yang *et al.*, 2021a] Jiancheng Yang, Rui Shi, and Bingbing Ni. Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis. In *ISBI*, pages 191–195, 2021.

[Yang *et al.*, 2021b] Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, and Bingbing Ni. Medmnist v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification. *arXiv preprint arXiv:2008.#TODO*, 2021.

[Yu *et al.*, 2020] Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. Salvaging federated learning by local adaptation. *arXiv preprint arXiv:2002.04758*, 2020.

[Yurochkin *et al.*, 2019] Mikhail Yurochkin, Mayank Agarwal, Soumya Ghosh, Kristjan Greenewald, Nghia Hoang, and Yasaman Khazaeni. Bayesian nonparametric federated learning of neural networks. In *ICML*, pages 7252–7261, 2019.

[Zaccone *et al.*, 2022] Riccardo Zaccone, Andrea Rizzardi, Debora Caldarola, Marco Ciccone, and Barbara Caputo. Speeding up heterogeneous federated learning with sequentially trained superclients. *arXiv*, 2022.

[Zeng *et al.*, 2022] Shenglai Zeng, Zonghang Li, Hongfang Yu, Yihong He, Zenglin Xu, Dusit Niyato, and Han Yu. Heterogeneous federated learning via grouped sequential-to-parallel training. *arXiv*, 2022.