




# HPFL: hyper-network guided personalized federated learning for multi-center tuberculosis chest x-ray diagnosis

Chang Liu<sup>1,2,3,4</sup> · Yong Luo<sup>1,2,3,4</sup> · Yongchao Xu<sup>1,2,3,4</sup> · Bo Du<sup>1,2,3,4</sup> 

Received: 19 June 2023 / Revised: 8 September 2023 / Accepted: 19 September 2023 /  
Published online: 23 October 2023

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

## Abstract

Tuberculosis (TB) is one of the widespread infectious disease, and the early diagnosis and treatment can greatly improve the survival rate. Recently, machine learning has been introduced for assisting the diagnosis of TB, and to train a reliable diagnosis model, we need large amounts of data, which are often distributed in multiple medical centers. To protect the data privacy of different centers, we introduce federated learning (FL) in tuberculosis diagnosis. Since the data distributions of TB data vary significantly across different centers, we propose a personalized FL (PFL) method to explore the specific property of each client (i.e., medical center), and reduce its negative impacts from other clients. In particular, the contribution of each layer parameter is quantified by a hyper-network customized by the server for each client. Besides, a parameterization mechanism is introduced to update the hierarchical aggregation weights. To the best of our knowledge, this is the first PFL method for distributed TB diagnosis. Experimental results on several public datasets of chest X-ray images show that the proposed method significantly outperforms the state-of-the-art approaches in terms of both higher accuracy and faster convergence speed.

**Keywords** Hyper-network · Federated learning · Tuberculosis · Chest x-ray

---

✉ Bo Du  
dubo@whu.edu.cn; remoteking@whu.edu.cn

Chang Liu  
computerscience@whu.edu.cn

Yong Luo  
luoyong@whu.edu.cn

Yongchao Xu  
yongchao.xu@whu.edu.cn

<sup>1</sup> School of Computer Science, Wuhan University, Wuhan 430072, China

<sup>2</sup> National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan 430072, China

<sup>3</sup> Institute of Artificial Intelligence, Wuhan University, Wuhan 430072, China

<sup>4</sup> Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan 430072, China

# 1 Introduction

Tuberculosis (TB) is one of the major global health threats, and the second leading cause of death from infectious diseases (after COVID-19) [1, 2]. There are around 10.6 million new TB patients diagnosed in 2021, and nearly 2 million of them died from TB [2]. Early diagnosis of TB and administration of antibiotic therapy can greatly improve the chance of survival [3–5]. Early screening using the chest X-ray medical image, as one of the most commonly used method for TB diagnosis, is important for the early detection, treatment and prevention of TB [1, 2, 5, 6]. However, radiologists often make errors on chest radiographs since it is often difficult for the human eye to quickly distinguish TB areas from other normal areas [1, 2]. According to [7], experienced radiologists only have an accuracy of 68.7% when the TB area is not obvious, while such area is critical for early diagnosis.

Besides, manual screening for large amounts of chest X-ray images is labor-intensive and time-consuming, and thus machine learning is introduced to assist the diagnosis of TB. Recently, due to the powerful representation capability, deep learning (DL) has been demonstrated to greatly improve diagnosis performance. Training a reliable deep learning model often requires large amounts of labeled data, which are hard to collect from a single medical center, and it is desirable to utilize data from multiple centers. However, due to privacy concerns about medical data, we may not be allowed to simply aggregate (merge) data across medical centers. Therefore, federated learning (FL) [8] is introduced to train a unified model using data from multiple centers without accessing to their original data. In FL, clients (i.e., medical centers in this paper) can keep their data private and share only the weights or gradients for model updates. Due to the privacy protection property, FL has been successfully applied to various tasks in medical image analysis [9–11].

A major challenge of FL in real-world applications is that the data are often non-IID (non-independent identically distributed) across different clients. This may lead to slow convergence and very poor inference performance when directly applying the learned global model to each client. This motivates the personalized federated learning (PFL), which allows each client to train a model adapted to local data [8]. However, existing PFL approaches usually simulate federated scenarios by regarding different groups of data in a single dataset as the different clients. The heterogeneity between different groups are often not that large compared with the real-world medical scenario, where the data properties and distributions of different medical centers may vary significantly.

There exists some tries that apply personalized federated learning for specific medical applications, such as federated domain adaptation for fMRI classification [9], federated normalization for pathology section staining [11], and federated contrast re-localization for CMR segmentation [10]. However, these approaches can only perform well for certain datasets and under certain settings (see our experiments). Besides, for datasets with large heterogeneity and small data volume, the existing PFL approaches may be inferior to the training using only local data. To this end, we propose a novel PFL method termed hyper-network guided personalized federated learning (HPFL) for distributed TB diagnosis, inspired by the dynamic models based on parameter isolation in continuous learning [12–14].

Specifically, our method consists of two main components: 1) To address the challenge of large heterogeneity across clients, the contribution of each layer parameter is quantified by a hyper-network customized by the server for each client. This allows clients to make truly beneficial updates to their own local models and thus enables each client to cope with possible catastrophic forgetting caused by the parameters of other clients that have very different data distributions; 2) To address the challenge of limited data in a single center and for many

atypical symptoms, a parameterization mechanism is introduced to update the hierarchical aggregation weights. This allows each client to gradually exploit the complementarity of different data among clients to reduce the requirements of data and communication overhead for model updates and thus accelerate the convergence.

To summarize, the main contributions of this paper are:

- We propose a novel PFL method for distributed TB diagnosis. To the best of our knowledge, this is the first PFL work for TB diagnosis, where the data heterogeneity issue is particularly served;
- We design hyper-networks to demonstrate the advantages of layer-wise aggregation over model-wise approaches in PFL among heterogeneous FL clients;
- We validate the effectiveness of our method on multiple real-world medical datasets and achieve better performance in terms of all metrics than some recent and competitive PFL counterparts.

## 2 Related work

### 2.1 Computer-aided tuberculosis diagnosis (CTD)

Owing to the sensitive nature of TB data and the difficulty in diagnosing TB using a gold standard, the availability of publicly accessible TB datasets is severely restricted [7]. Due to the scarcity of data, conventional CTD methods are inadequate to train deep Convolutional Neural Networks (CNNs). Traditional techniques primarily rely on hand-crafted features and binary classifiers [15]. Jaeger et al. [5] proposed a graph cut segmentation method [16] to segment the lung area. They then extracted hand-crafted texture and shape features from the segmented region and employed a binary classifier, i.e., support vector machine (SVM), to differentiate normal and abnormal X-rays. Candemir et al. [6] utilized patient-specific adaptive lung models based on image retrieval and a nonrigid registration-driven robust lung segmentation method to facilitate traditional lung feature extraction. Chauhan et al. [3] designed a MATLAB toolbox, TB-Xpredict, which used Gist [17] and PHOG [18] features for discriminating between TB and non-TB X-rays without requiring segmentation. Karargyris et al. [19] extracted shape features to define the overall geometrical characteristics of lungs and texture features to represent image characteristics.

Contrary to using hand-crafted features, Lopes et al. [20] employed the pre-trained fixed CNNs from ImageNet [21] as the feature extractors to generate deep features for X-ray images. They then employed SVM to classify these deep features. Hwang et al. [4] trained an AlexNet for binary classification (TB and non-TB) using a private dataset. Rajaraman and Antani [22] proposed a methodology for explicit collective learning to enhance abnormality detection in CXRs. The approach involved combining model predictions using various ensemble strategies to reduce prediction variance and sensitivity to the training data, thereby improving overall performance and generalization.

### 2.2 Federated learning with non-IID data

The standard setup of FL seeks to train a single global model that can perform well on generic data distributions [23]. As clients' data are kept separate, mainstream algorithms like FedAvg [8] take a multi-round approach. Within each round, the server first broadcasts the global model to the clients, who then independently update it locally using their own

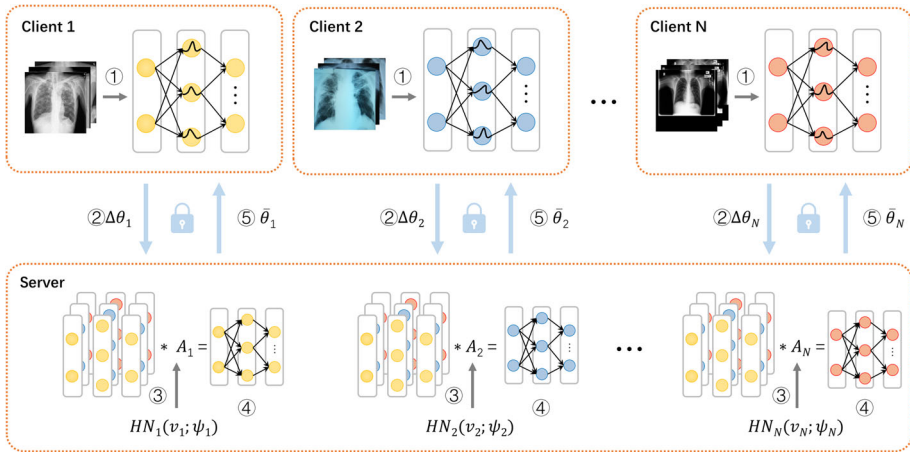
(often limited) data. The server then aggregates the local models into the global model and proceeds to the next round. This pipeline is shown promising if clients' data are IID (i.e., with similar data and label distributions), which is, however, hard to meet in reality and thus results in a performance drop. Several approaches have been proposed to enhance its performance. Some studies concentrate on optimizing local learning algorithms by leveraging well-designed objective regularization and local bias correction. For instance, FedProx [24] introduces a proximal term to the local training objective, ensuring that updated parameters remain close to the original downloaded model. However, in medical image analysis, the target distributions between medical institutions often suffer non-IID data problems, and a single global model may not be adequate [25]. In standard federated learning, due to the utilization of a shared global model across all clients, achieving desirable results in non-IID scenarios can be challenging without careful adjustments. Conversely, personalized federated learning, which allows for customized models at each local client, often exhibits greater robustness in non-IID scenarios.

### 2.3 Personalized federated learning

Several popular personalized FL methods can be found in the literature, including multi-task learning with model dissimilarity penalization, such as Ditto [26], and parameter decoupling of feature extractor and classifier, such as FedPer [27] and FedRep [28]. However, existing client-specific FL methods are often developed by heuristically evaluating model similarity or validation accuracy and must balance communication/computation overhead with personalization effectiveness. FedBN discovered that local models with Batch Norm layers could exclude these parameters from the aggregating steps during training, reducing communication costs while enhancing personalization [29]. More recently, FedBABU maintains the global classifier unchanged during feature representation learning and performs local adaptation by fine-tuning [30]. FedRoD proposed using a balanced softmax for learning a generic model and vanilla softmax for personalized heads [31]. However, these literatures has yet to consider the layer-wised utility for personalized aggregation. The distance metric used to describe the similarity among models can be inaccurate and may lead to sub-optimal performance. This motivates us to explore a fine-grained aggregation strategy to adapt to a broad range of non-IID clients.

### 2.4 Hyper-networks

Hyper-networks [32] are utilized to generate model parameters for other neural networks, such as a target network, by mapping the embeddings of the target tasks to corresponding parameters. Hyper-networks have found extensive use in diverse machine learning applications, including language modeling [33, 34], computer vision [35, 36], 3D scene representation [37, 38], hyperparameter optimization [39–42], and meta-learning [43]. Shamsian et al. [44] were the first to apply hyper-networks in FL, enabling the generation of personalized and effective model parameters for each client. Ma et al. [45] demonstrate that hyper-networks can evaluate the importance of individual model layers and enhance personalized aggregation in non-IID scenarios (Figs. 1, 2 and 3).

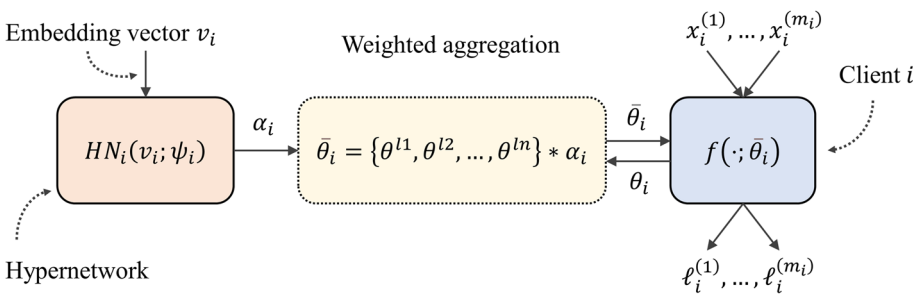


**Fig. 1** Framework of our HPFL, which mainly consists of five steps: 1) local training using private data; 2) each client sends the update of parameters  $\Delta\theta_i$  to the server; 3) the server updates the aggregation weight matrix  $\alpha_i$  by hyper-networks  $HN_i(v_i; \psi_i)$  according to  $\Delta\theta_i$ ; 4) the server performs weighted aggregation and outputs personalized model  $\bar{\theta}_i$  for the corresponding client; 5) the server sends the personalized model  $\bar{\theta}_i$  to each client

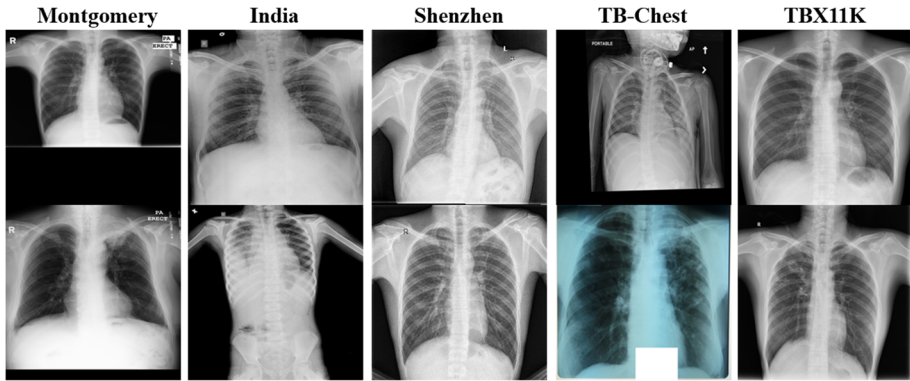
### 3 Methodology

#### 3.1 PFL formulation

Personalized federated learning (PFL) aims to collaboratively train personalized models for a set of  $N$  clients, each with its own personal private data  $m_i$ . Unlike conventional FL, each client  $i$  is equipped with its own data distribution  $\mathcal{P}_i$  on  $\mathcal{X} \times \mathcal{Y}$ . Assume each client has access to  $m_i$  samples from  $\mathcal{P}_i$ ,  $\mathcal{S}_i = \left\{ \left( \mathbf{x}_j^{(i)}, y_j^{(i)} \right) \right\}_{j=1}^{m_i}$ . Let  $\ell_i : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+$  denote the loss function corresponds to client  $i$ , and  $\mathcal{L}_i$  the average loss over the personal training data  $\mathcal{L}_i(\theta_i) = \frac{1}{m_i} \sum_j \ell_i(\mathbf{x}_j, y_j; \theta_i)$ . Here  $\theta_i$  denotes the personal model of client  $i$ . The objective



**Fig. 2** Illustration of one hypernetwork framework used in HPFL. The hypernetwork  $HN_i$  takes the embedding vector  $v_i$  as input, and outputs the aggregation weight matrix  $\alpha_i$ . After the weighted combination with intermediate parameters  $\{\theta^{l1}, \dots, \theta^{ln}\}$  and aggregation weight matrix  $\alpha_i$ , client  $i$  can make local training on private data. Note that both  $v_i$  and  $\psi_i$  are updated during training



**Fig. 3** A randomly selected sample from each data set. Due to the differences in collection devices, collection subjects, and collection methods, there are significant differences in the data from different datasets

of PFL is to optimize

$$\Theta^* = \arg \min_{\Theta} \sum_{i=1}^N \frac{m_i}{M} \mathcal{L}_i(\theta_i), \quad (1)$$

where  $\Theta = \{\theta_1, \dots, \theta_N\}$  is the set of personalized parameters for all clients.  $M$  is the total amount of data in all clients.  $m_i$  is the amount of data owned by the  $i$ -th client.  $\mathcal{L}_i$  is loss function of  $i$ -th client associated with dataset  $\mathcal{D}_i$ .

### 3.2 The proposed HPFL

In this section, we present our proposed PFL algorithm HPFL, which evaluates the importance of each layer from different clients to achieve layer-wised personalized model aggregation. We apply a dedicated hyper-network for each client on the server and train them to generate aggregation weights for each model layer of different clients. Unlike the general FL framework that generates only one global model, HPFL maintains a personalized model for each client at the server. Clients with similar data distribution should have high aggregation weights to reinforce the mutual contribution from each other. This approach can expedite model convergence and effectively reduce the computational burden on the client side. This is particularly advantageous for deployment in economically underdeveloped regions, where the prevalence of tuberculosis tends to be higher due to economic backwardness [7].

Our HPFL applies a set of aggregation weight matrix  $\alpha_i$  at the server side to progressively exploit the inter-user similarities at the layer level, which is defined as

$$\alpha_i = [\alpha_i^{l1}, \alpha_i^{l2}, \dots, \alpha_i^{ln}] = \begin{bmatrix} \alpha_i^{l1,1} & \alpha_i^{l2,1} & \dots & \alpha_i^{ln,1} \\ \alpha_i^{l1,2} & \alpha_i^{l2,2} & \dots & \alpha_i^{ln,2} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_i^{l1,N} & \alpha_i^{l2,N} & \dots & \alpha_i^{ln,N} \end{bmatrix}, \quad (2)$$

where  $\alpha_i^{ln}$  represents the aggregation weight vector of  $n$ -th layer in client  $i$ , while  $\alpha_i^{ln,N}$  represents the aggregation weight for client  $N$  in  $n$ -th layer. For all  $n$  layers,  $\sum_{j=1}^N \alpha_i^{ln,j} = 1$ .

Different from previous PFL algorithms, instead of applying identical weight values for all layers of a client model, HPFL considers the different utilities of neural layers and assigns a

unique weight to each of them to achieve fine-grained personalized aggregation. In addition, unlike traditional methods that mathematically calculate the weights using a distance metric among the entire model parameters, HPFL parameterized the weights during the training phase via a set of dedicated hyper-networks. The layer-wised weights are determined by the hyper-networks, which are alternatively updated with the personalized model. In such way, we can obtain effective weights as their update direction is in line with the optimization direction of the objective function. In the following, we will elaborate on the updating process of the aggregation weight matrix  $\alpha$  of HPFL.

Each hyper-network consists of several fully connected layers, whose input is an embedding vector that is automatically updated with the model parameters, and the output is the weight matrix  $\alpha$ . Define the hyper-network on client  $i$  as

$$\alpha_i = HN_i(v_i; \psi_i), \tag{3}$$

where  $v_i$  is the embedding vector and  $\psi_i$  is the parameter of client  $i$ 's hyper-network. Let  $\{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\}$  be the intermediate parameters of all clients after local training,  $\theta^{ln} = \{\theta_1^{ln}, \theta_2^{ln}, \dots, \theta_N^{ln}\}$  is the set of  $n$ -th layer of all clients, where  $\theta_N^{ln}$  are the parameters of  $n$ -th layer in client  $N$ . In HPFL, the model parameters of client  $i$  is obtained by weighted aggregation according to  $\alpha_i$ , i.e.,

$$\bar{\theta}_i = \{\bar{\theta}_i^{l1}, \bar{\theta}_i^{l2}, \dots, \bar{\theta}_i^{ln}\} = \{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\} * \alpha_i, \tag{4}$$

where  $\bar{\theta}_i^{ln}$  can also be expressed as:

$$\bar{\theta}_i^{ln} = \sum_{j=1}^N \theta_j^{ln} \alpha_i^{ln,j} \tag{5}$$

Thus, the objective function of our HPFL can be derived by reformulated (1) as

$$\arg \min_{V, \Psi} \sum_{i=1}^N \frac{m_i}{M} \mathcal{L}_i \left( \{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\} * HN_i(v_i; \psi_i) \right), \tag{6}$$

where  $V = \{v_1, \dots, v_N\}$ ,  $\Psi = \{\psi_1, \dots, \psi_N\}$ . Consequently, HPFL transforms the optimization problem for client parameters  $\theta_i$  into the hypernetwork  $k$ 's embedding vector  $v_i$  and parameters  $\psi_i$ . In the following, we introduce the updated rules of  $V$  and  $\Psi$ .

### 3.2.1 Update $v_i$ and $\psi_i$

Inspired by [44], using the chain rule we have  $\nabla_{v_i} \mathcal{L}_i = (\nabla_{v_i} \bar{\theta}_i)^T \nabla_{\bar{\theta}_i} \mathcal{L}_i$ , and we can have the gradient of  $v_i$  and  $\psi_i$  from (6):

$$\nabla_{v_i} \mathcal{L}_i = (\nabla_{v_i} \bar{\theta}_i)^T \nabla_{\bar{\theta}_i} \mathcal{L}_i = \left[ \{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\} * \nabla_{v_i} HN_i(v_i; \psi_i) \right]^T \nabla_{\bar{\theta}_i} \mathcal{L}_i, \tag{7}$$

$$\nabla_{\psi_i} \mathcal{L}_i = (\nabla_{\psi_i} \bar{\theta}_i)^T \nabla_{\bar{\theta}_i} \mathcal{L}_i = \left[ \{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\} * \nabla_{\psi_i} HN_i(v_i; \psi_i) \right]^T \nabla_{\bar{\theta}_i} \mathcal{L}_i, \tag{8}$$

where  $\nabla_{\bar{\theta}_i} \mathcal{L}_i$  can be obtained from client  $i$ 's local training in each communication round and  $\nabla_{v_i/\psi_i} HN_i(v_i; \psi_i)$  is the gradient of  $\alpha_i$  in directions  $v_i/\psi_i$ . We use a more general way to update  $v_i$  and  $\psi_i$ , i.e.,

$$\Delta v_i = (\nabla_{v_i} \bar{\theta}_i)^T \Delta \theta_i = \left[ \{\theta^{l1}, \theta^{l2}, \dots, \theta^{ln}\} * \nabla_{v_i} HN_i(v_i; \psi_i) \right]^T \Delta \theta_i, \tag{9}$$

$$\Delta\psi_i = (\nabla_{\psi_i} \bar{\theta}_i)^T \Delta\theta_i = \left[ \left\{ \theta^{l1}, \theta^{l2}, \dots, \theta^{ln} \right\} * \nabla_{\psi_i} H N_i (v_i; \psi_i) \right]^T \Delta\theta_i, \quad (10)$$

where  $\Delta\theta_i$  is the change of model parameters in client  $i$  after local training. In accordance with (9) and (10), HPFL updates the embedding vector and parameters of hypernetwork for client  $i$  at each communication round, and then update the aggregation weight matrix  $\alpha_i$ .

In order to facilitate more efficient convergence, the parameters of models are initialized using pre-trained weights from ImageNet. In each communication round, the clients first download the latest personalized models from the server, and then train several epochs based on the private data. After that, the model update  $\Delta\theta_i$  for each client will be uploaded to the server to update the embedding vector  $V$  and the parameter  $\Psi$ .

## 4 Experiments

### 4.1 Datasets

To validate the performance of HPFL on multi-center diagnosis, we construct a real-world FL benchmark from publicly available datasets. Those datasets are collected in different countries (US, China, India, Qatar and Bangladesh) different years. The details are illustrated in Table 1. For the real challenge with a small amount of golden standard data, in addition to the routine experiments of using 80% training and 20% testing, we also conducted experiments using 20% training and 80% testing.

### 4.2 Implementation details

We conducted several experiments based on ResNet50 [48], Swin Transformer-tiny [49], ConvNeXt V2-A [50], and finally chose ConvNeXt V2-A, which is both lightweight and stable, for the more detailed experimental analysis later (see Table 2). The network is optimized using Adam with a batch size of 32. The total communication rounds is 300, with the local training epoch set as 1. The learning rate is reduced by half every 50 rounds. Unlike the common federated setting of dividing multiple clients in a dataset, we set each dataset as a client in order to follow the real medical scenario. Since 5 datasets are used, 5 clients are naturally formed. Since different federated learning algorithms have different optimal

**Table 1** Chest X-ray datasets used in our experiments

Dataset	Year	Samples of all	Samples of TB
Montgomery [46]	2014	138	58
India [3]	2014	278	125
Shenzhen [46]	2014	662	336
TB-Chest [47]	2020	4200	700
TBX11K [7]	2020	11200	800*

\*In TBX11K, only the training set has labels for 800 TB samples. Due to the difficulty in obtaining the gold standard for TB and the high degree of privacy involved, the data volume of commonly used public datasets is currently small.



**Table 2** Comparison of micro-average performance using different backbones

Method	ResNet50 [48]		SwinT-Tiny [49]		ConvNeXt-V2A [50]		Average	
	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
Local-only	96.26	94.82	96.12	95.14	97.98	96.61	96.79	95.52
FedAvg [8]	96.60	94.43	97.73	95.18	98.46	96.51	97.60	95.37
AISTATS'17								
FedPer [27]	97.45	95.67	98.17	96.58	98.71	96.37	98.11	96.21
arXiv'19								
FedBN [29]	96.65	95.00	97.41	94.96	97.88	95.99	97.31	95.32
ICLR'21								
Ditto [26]	96.61	94.42	97.57	94.47	98.41	96.37	97.53	95.09
ICML'21								
FedRep [28]	96.77	94.75	97.55	95.53	97.09	93.57	97.14	94.62
ICML'21								
FedBABU [30]	95.85	92.61	96.84	94.57	98.60	96.68	97.10	94.62
ICLR'22								
FedRoD [31]	96.18	92.08	96.80	90.77	97.36	92.08	96.78	91.64
ICLR'22								
HPFL (ours)	<b>98.16</b>	<b>96.70</b>	<b>98.57</b>	<b>97.33</b>	<b>99.07</b>	<b>97.60</b>	<b>98.60</b>	<b>97.21</b>

80% of the data was used for training and 20% for testing

**Table 3** Results of AUC comparison with advanced methods

Method	Montgomery	India	Shenzhen	TB-Chest	TBX11K	Average
Local-only	46.47	91.81	88.62	99.86	98.63	85.08
	± 2.68	± 0.63	± 0.04	± 0.05	± 1.12	± 0.81
FedAvg [8]	84.83	83.83	92.14	99.80	99.83	92.09
AISTATS'17	± 1.68	± 1.84	± 0.36	± 0.07	± 0.04	± 0.68
FedPer [27]	84.38	85.45	91.66	99.64	99.80	92.18
arXiv'19	± 2.16	± 0.80	± 1.56	± 0.14	± 0.05	± 0.71
FedBN [29]	84.49	79.33	88.00	99.56	99.78	90.23
ICLR'21	± 2.19	± 2.42	± 1.59	± 0.11	± 0.07	± 1.08
Ditto [26]	85.28	85.43	89.46	99.70	99.79	91.93
ICML'21	± 1.03	± 1.69	± 2.29	± 0.09	± 0.07	± 1.00
FedRep [28]	77.69	81.27	86.13	98.33	98.93	88.47
ICML'21	± 2.70	± 2.19	± 0.86	± 0.03	± 0.16	± 0.76
FedBABU [30]	84.33	84.15	91.49	99.71	99.76	91.89
ICLR'22	± 3.00	± 0.42	± 1.07	± 0.07	± 0.04	± 0.88
FedRoD [31]	75.47	74.38	80.76	98.51	99.23	85.67
ICLR'22	± 0.92	± 1.76	± 0.90	± 0.07	± 0.10	± 0.55
HPFL (ours)	<b>87.81</b>	<b>92.94</b>	<b>92.23</b>	<b>99.87</b>	<b>99.84</b>	<b>94.54</b>
	± 0.94	± 0.75	± 0.20	± 0.03	± 0.02	± 0.31

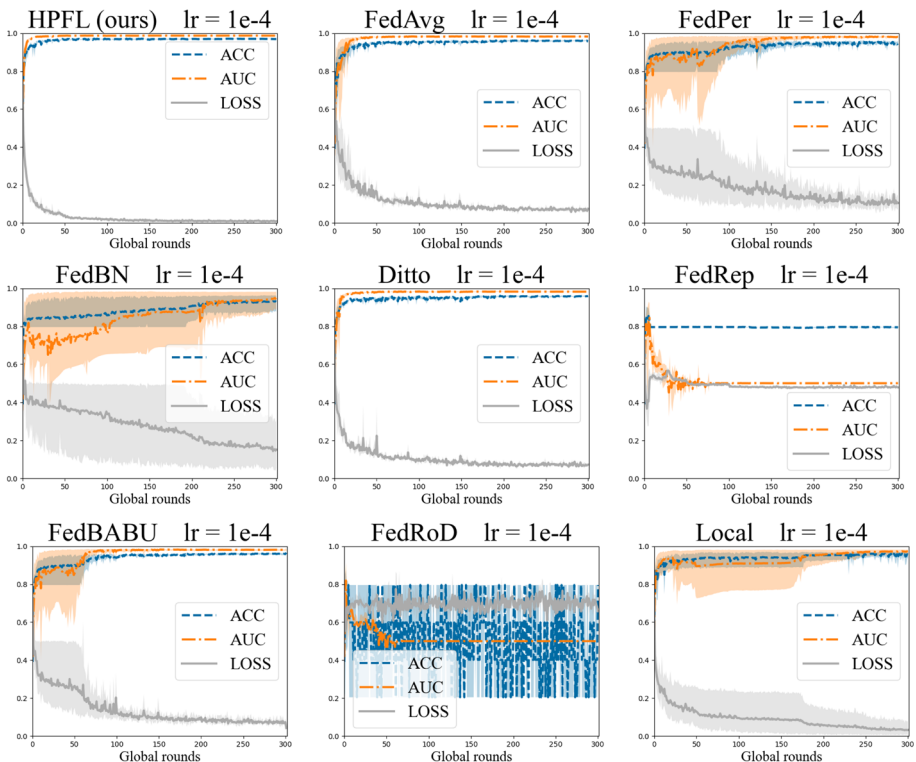
To address the challenge of scarce gold standard data in real scenarios, only 20% of the data is used for training, and the remaining 80% is used for testing. Experiments for each method were repeated three times

learning rates, we compared four initial learning rate settings of  $1e-3$ ,  $1e-4$ ,  $1e-5$ , and  $1e-6$  for the sake of fairness of the experiments.

### 4.3 Experimental analysis

Comparing Tables 2 and 3, we can see that the larger the test set relative to the training set, i.e., the more data not seen in training, the more obvious the advantage of federated learning over training with only local data.

In whichever configuration, our PHFL has superior performance in multi-center TB diagnoses than the state-of-the-art PFL methods. It is interesting to note that among the various SOTA that emerged after FedAvg, only FedPer exceeded FedAvg in terms of average accuracy and AUC-ROC. We analyze that the reason for this result is that PFL methods are usually trained and tested with datasets such as CIFAR-10, CIFAR-100, Tiny-ImageNet, etc. These image datasets cannot be aligned due to the different label classes, so most PFL methods are tested by simulating the federated within each dataset. However, medical data in real scenarios, especially TB chest X-rays, have small differences within each dataset and large differences between datasets. On the one hand, since the weight of parameter updates is given according to the amount of data in federated learning, centers with limited data and large heterogeneity are easily carried away from the local distribution by centers with large amounts

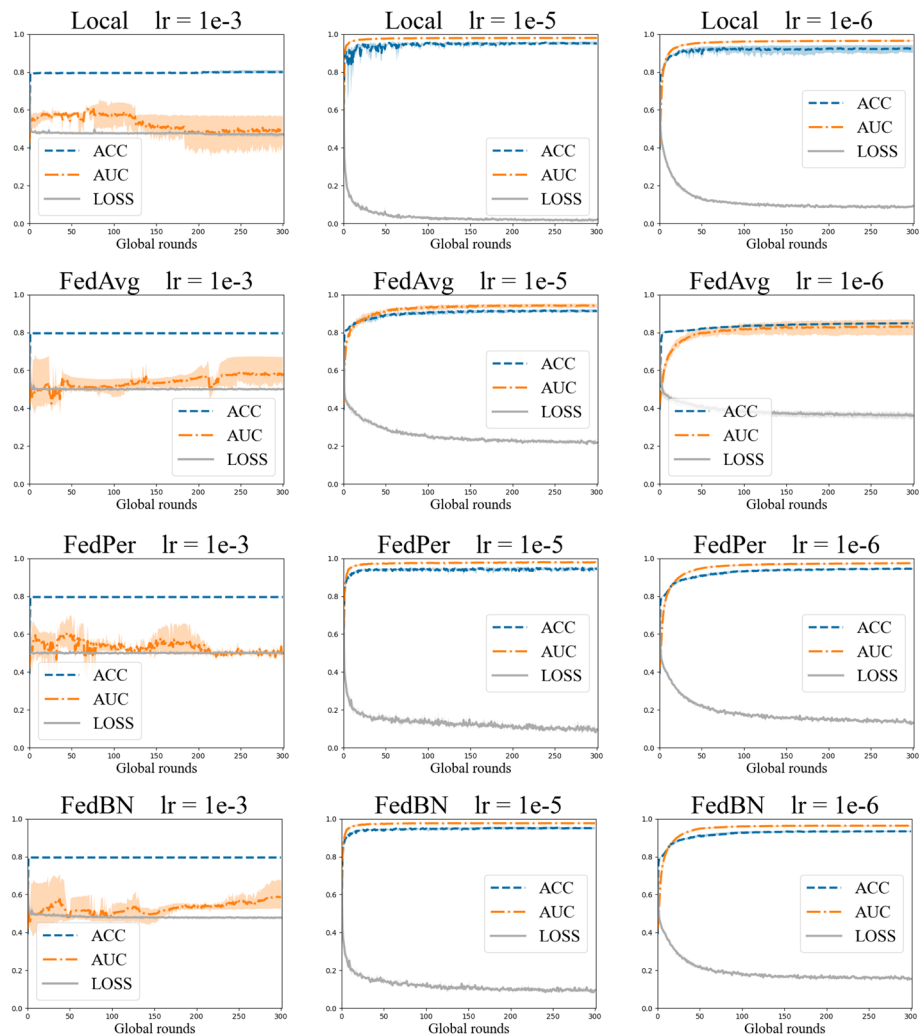


**Fig. 4** Under three times random seeds, most methods, including FedAvg, give the best results at a learning rate of  $1e-4$ , while our method demonstrates the best, fastest, and most stable convergence

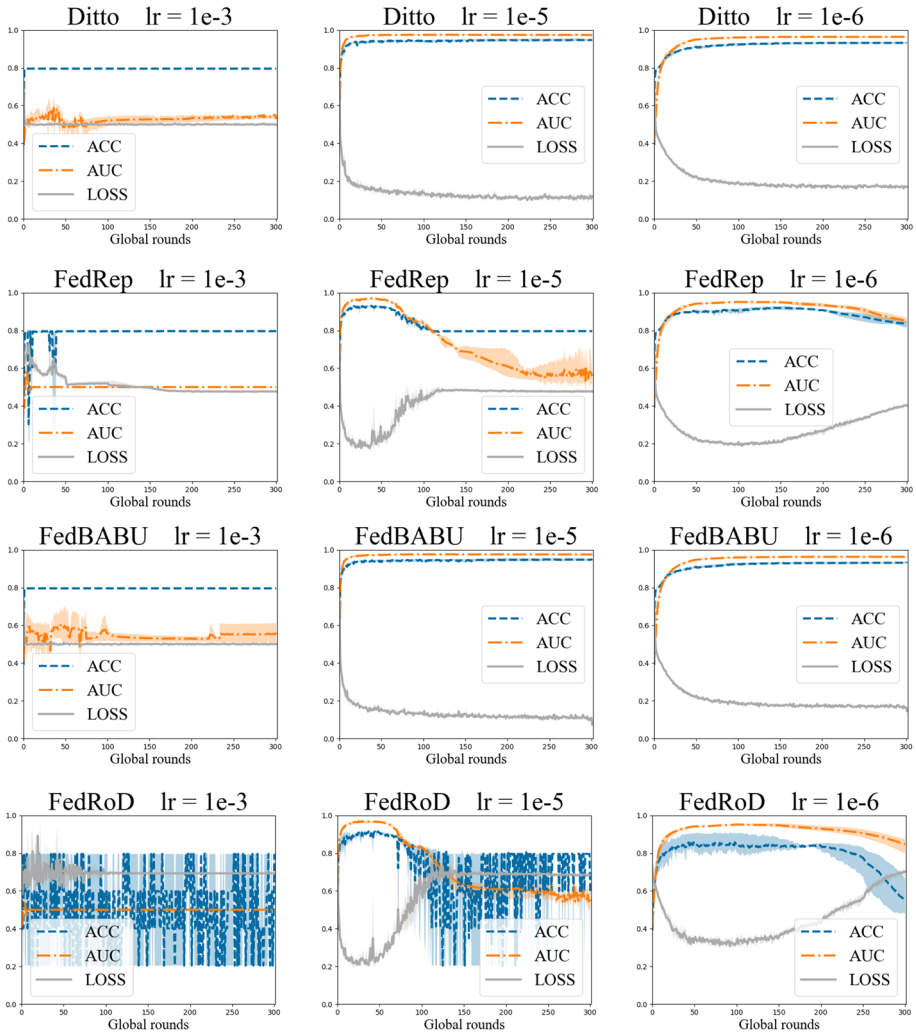
of data. On the other hand, small data sets can also lead to the overfitting of personalized federated methods when the local data distribution is not representative enough and when inappropriate personalization is performed.

It can be seen that different personalized federated methods have their own suitable dataset and settings, and there is no one constant aggregation and optimization method for all scenarios. And for datasets with large heterogeneity and small data volume, such as India, existing personalized federated methods bring negative optimization that is inferior to local training.

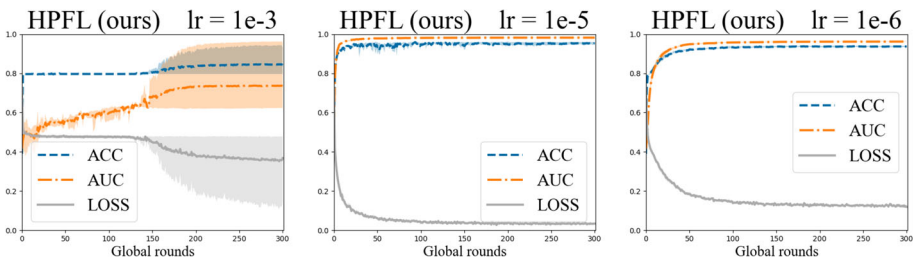
To deal with these problems, our approach quantifies the contribution of each layer parameter through a server-tailored hyper-network for each client, allowing each client to make truly beneficial updates to its own local layer granularity without being negatively impacted by the different data distribution of other centers. Learn the distribution of different datasets



**Fig. 5** As the learning rate decreases, the convergence of the various methods becomes more stable, but too low a learning rate can also lead to incomplete optimization



**Fig. 6** Some of the methods, including FedRep and FedRoD, are suitable for SGD optimization but not for Adam. But in practice, using only SGD optimization is not only slow to converge but also requires a lot of tuning work to achieve the desired optimization results



**Fig. 7** Compared to other methods, our method is optimal at almost all initial learning rates and is also more robust to excessive learning rates

through dynamic personalization. As can be seen from Tables 2 and 3, our approach yields optimal results regardless of the backbone and the dataset. The comparison of Figs. 4, 5 and 6 shows that most of the algorithms fail to converge when the learning rate is  $1e-3$ , but our method can be optimized adaptively at a later stage (Fig. 7).

## 5 Conclusion

Early diagnosis is very important for the treatment and prevention of TB, a major infectious disease. However, accurate diagnosis of TB remains a major challenge. Inspired by the success of deep learning, deep learning-based computer-aided tuberculosis diagnosis (CTD) has become a promising research direction, but the issue of data silos may lead to poor performance.

To address the challenge of large data heterogeneity, we propose a novel PFL method to explore the specific property of each client (i.e., medical center) and reduce its negative impacts on other clients. In particular, the contribution of each layer parameter is quantified by hyper-networks customized by the server for each client. From the extensive experiments on multiple real-world medical datasets, we mainly conclude that: 1) existing PFL approaches may perform worse than the simple FedAvg in the TB diagnosis, while our HPFL consistently outperforms FedAvg; 2) the personalized heads of higher layers often contribute more for each client, while the higher granularity aggregation method gives better results. In the future, we intend to design more sophisticated strategies for personalized parameter aggregation and update.

**Acknowledgements** This work was supported in part by the National Natural Science Foundation of China (Grant No. 62225113, 62222112, 62176186, and 62276195).

**Data Availability** The author confirms that all data generated or analysed during this study are included in this published article. Furthermore, primary and secondary sources and data supporting the findings of this study were all publicly available at the time of submission.

## Declarations

**Conflict of interests** The authors declare that they have no conflict of interest.

## References

1. WHO (2021) Global tuberculosis report 2021. 1–57. [https://www.who.int/publications/digital/global\\_tuberculosis-report-2021](https://www.who.int/publications/digital/global_tuberculosis-report-2021)
2. WHO (2022) Global tuberculosis report 2022. 1–68. [https://www.who.int/teams/global\\_tuberculosis-programme/tb-reports/global-tuberculosis-report-2022](https://www.who.int/teams/global_tuberculosis-programme/tb-reports/global-tuberculosis-report-2022)
3. Chauhan A, Chauhan D, Rout C (2014) Role of gist and phog features in computer-aided diagnosis of tuberculosis without segmentation. *PloS One* 9(11):112980
4. Hwang S, Kim H-E, Jeong J, Kim H-J (2016) A novel approach for tuberculosis screening based on deep convolutional neural networks. In: *Medical imaging 2016: computer-aided diagnosis*, vol 9785, pp 750–757. SPIE
5. Jaeger S, Karargyris A, Candemir S, Folio L, Siegelman J, Callaghan F, Xue Z, Palaniappan K, Singh RK, Antani S et al (2013) Automatic tuberculosis screening using chest radiographs. *IEEE Trans Med Imaging* 33(2):233–245

6. Candemir S, Jaeger S, Palaniappan K, Musco JP, Singh RK, Xue Z, Karargyris A, Antani S, Thoma G, McDonald CJ (2013) Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. *IEEE Trans Med Imaging* 33(2):577–590
7. Liu Y, Wu Y-H, Ban Y, Wang H, Cheng M-M (2020) Rethinking computer-aided tuberculosis diagnosis. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp 2646–2655
8. McMahan B, Moore E, Ramage D, Hampson S, Arcas BA (2017) Communication-efficient learning of deep networks from decentralized data. In: *Artificial intelligence and statistics*, pp 1273–1282. PMLR
9. Li X, Gu Y, Dvornek N, Staib LH, Ventola P, Duncan JS (2020) Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results. *Med Image Anal* 65:101765
10. Qi X, Yang G, He Y, Liu W, Islam A, Li S (2022) Contrastive re-localization and history distillation in federated cmr segmentation. In: *Medical image computing and computer assisted intervention—MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part V*, pp 256–265. Springer
11. Shen Y, Sowmya A, Luo Y, Liang X, Shen D, Ke J (2022) A federated learning system for histopathology image analysis with an orchestral stain-normalization gan. *IEEE Trans Med Imaging*
12. Lomonaco V, Maltoni D, Pellegrini L (2020) Rehearsal-free continual learning over small non-iid batches. In: *CVPR Workshops*, vol 1, pp 3
13. Serra J, Suris D, Miron M, Karatzoglou A (2018) Overcoming catastrophic forgetting with hard attention to the task. In: *International conference on machine learning*, pp 4548–4557. PMLR
14. Wortsman M, Ramanujan V, Liu R, Kembhavi A, Rastegari M, Yosinski J, Farhadi A (2020) Supermasks in superposition. *Adv Neural Inf Process Syst* 33:15173–15184
15. Santosh K, Allu S, Rajaraman S, Antani S (2022) Advances in deep learning for tuberculosis screening using chest x-rays: The last 5 years review. *J Med Syst* 46(11):82
16. Boykov Y, Funka-Lea G (2006) Graph cuts and efficient nd image segmentation. *Int J Comput Vis* 70(2):1–2
17. Oliva A, Torralba A (2006) Building the gist of a scene: The role of global image features in recognition. *Prog Brain Res* 155:23–36
18. Bosch A, Zisserman A, Munoz X (2007) Representing shape with a spatial pyramid kernel. In: *Proceedings of the 6th ACM international conference on image and video retrieval*, pp 401–408
19. Karargyris A, Siegelman J, Tzortzis D, Jaeger S, Candemir S, Xue Z, Santosh K, Vajda S, Antani S, Folio L et al (2016) Combination of texture and shape features to detect pulmonary abnormalities in digital chest x-rays. *Int J Comput Assist Radiol Surg* 11:99–106
20. Lopes U, Valiati JF (2017) Pre-trained convolutional neural networks as feature extractors for tuberculosis detection. *Comput Biol Med* 89:135–143
21. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: A large-scale hierarchical image database. In: *2009 IEEE Conference on computer vision and pattern recognition*, pp 248–255. IEEE
22. Rajaraman S, Kim I, Antani SK (2020) Detection and visualization of abnormality in chest radiographs using modality-specific convolutional neural network ensembles. *PeerJ* 8:8693
23. Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Bhagoji AN, Bonawitz K, Charles Z, Cormode G, Cummings R et al (2021) Advances and open problems in federated learning. *Found Trends Mach Learn* 14(1–2):1–210
24. Li X, Huang K, Yang W, Wang S, Zhang Z (2019) On the convergence of fedavg on non-iid data. [arXiv:1907.02189](https://arxiv.org/abs/1907.02189)
25. Fallah A, Mokhtari A, Ozdaglar A (2020) Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Adv Neural Inf Process Syst* 33:3557–3568
26. Li T, Hu S, Beirami A, Smith V (2021) Ditto: Fair and robust federated learning through personalization. In: *International conference on machine learning*, pp 6357–6368. PMLR
27. Arivazhagan MG, Aggarwal V, Singh AK, Choudhary S (2019) Federated learning with personalization layers. [arXiv:1912.00818](https://arxiv.org/abs/1912.00818)
28. Collins L, Hassani H, Mokhtari A, Shakkottai S (2021) Exploiting shared representations for personalized federated learning. In: *International conference on machine learning*, pp 2089–2099. PMLR
29. Li X, Jiang M, Zhang X, Kamp M, Dou Q (2021) Fedbn: Federated learning on non-iid features via local batch normalization. [arXiv:2102.07623](https://arxiv.org/abs/2102.07623)
30. Oh J, Kim S, Yun S-Y (2021) Fedbabu: Towards enhanced representation for federated image classification. [arXiv:2106.06042](https://arxiv.org/abs/2106.06042)
31. Chen H-Y, Chao W-L (2021) On bridging generic and personalized federated learning for image classification. [arXiv:2107.00778](https://arxiv.org/abs/2107.00778)
32. Ha D, Dai A, Le QV (2016) Hypernetworks. [arXiv:1609.09106](https://arxiv.org/abs/1609.09106)

33. Nirkin Y, Wolf L, Hassner T (2021) Hyperseg: Patch-wise hypernetwork for realtime semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 4061–4070
34. Suarez J (2017) Language modeling with recurrent highway hypernetworks. *Adv Neural Inf Process Syst* 30
35. Jia X, De Brabandere B, Tuytelaars T, Gool LV (2016) Dynamic filter networks. *Adv Neural Inf Process Syst* 29
36. Littwin E, Galanti T, Wolf L, Yang G (2020) On infinite-width hypernetworks. *Adv Neural Inf Process Syst* 33:13226–13237
37. Littwin G, Wolf L (2019) Deep meta functionals for shape representation. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 1824–1833
38. Sitzmann V, Martel J, Bergman A, Lindell D, Wetzstein G (2020) Implicit neural representations with periodic activation functions. *Adv Neural Inf Process Syst* 33:7462–7473
39. Bae J, Grosse RB (2020) Delta-stn: Efficient bilevel optimization for neural networks using structured response jacobians. *Adv Neural Inf Process Syst* 33:21725–21737
40. Li Y, Gu S, Zhang K, Van Gool L, Timofte R (2020) Dhpr: Differentiable meta pruning via hypernetworks. In: Computer vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VIII 16, pp 608–624. Springer
41. Lorraine J, Duvenaud D (2018) Stochastic hyperparameter optimization through hypernetworks. [arXiv:1802.09419](https://arxiv.org/abs/1802.09419)
42. MacKay M, Vicol P, Lorraine J, Duvenaud D, Grosse R (2019) Self-tuning networks: Bilevel optimization of hyperparameters using structured best-response functions. [arXiv:1903.03088](https://arxiv.org/abs/1903.03088)
43. Zhao D, Oswald J, Kobayashi S, Sacramento J, Grewe BF (2020) Meta-learning via hypernetworks
44. Shamsian A, Navon A, Fetaya E, Chechik G (2021) Personalized federated learning using hypernetworks. In: International conference on machine learning, pp 9489–9502. PMLR
45. Ma X, Zhang J, Guo S, Xu W (2022) Layer-wised model aggregation for personalized federated learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 10092–10101
46. Jaeger S, Candemir S, Antani S, Wang Y-XJ, Lu P-X, Thoma G (2014) Two public chest x-ray datasets for computer-aided screening of pulmonary diseases. *Quant Imaging Med Surg* 4(6):475
47. Rahman T, Khandakar A, Kadir MA, Islam KR, Islam KF, Mazhar R, Hamid T, Islam MT, Kashem S, Mahbub ZB et al (2020) Reliable tuberculosis detection using chest x-ray with deep learning, segmentation and visualization. *IEEE Access* 8:191586–191601
48. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
49. Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, Lin S, Guo B (2021) Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 10012–10022
50. Woo S, Debnath S, Hu R, Chen X, Liu Z, Kweon IS, Xie S (2023) Convnext v2: Co-designing and scaling convnets with masked autoencoders. [arXiv:2301.00808](https://arxiv.org/abs/2301.00808)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.