FedTrip: A Resource-Efficient Federated Learning Method with Triplet Regularization

Xujing Li*[†], Min Liu*[†], Sheng Sun*, Yuwei Wang*, Hui Jiang*[†], Xuefeng Jiang*[†]

*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

[†]University of Chinese Academy of Sciences, Beijing, China

*{lixujing19b, liumin, sunsheng, ywwang, jianghui, jiangxuefeng21b}@ict.ac.cn,

Abstract—In the federated learning scenario, geographically distributed clients collaboratively train a global model. Data heterogeneity among clients significantly results in inconsistent model updates, which evidently slow down model convergence. To alleviate this issue, many methods employ regularization terms to narrow the discrepancy between client-side local models and the server-side global model. However, these methods impose limitations on the ability to explore superior local models and ignore the valuable information in historical models. Besides, although the up-to-date representation method simultaneously concerns the global and historical local models, it suffers from unbearable computation cost. To accelerate convergence with low resource consumption, we innovatively propose a model regularization method named FedTrip, which is designed to restrict global-local divergence and decrease current-historical correlation for alleviating the negative effects derived from data heterogeneity. FedTrip helps the current local model to be close to the global model while keeping away from historical local models, which contributes to guaranteeing the consistency of local updates among clients and efficiently exploring superior local models with negligible additional computation cost on attaching operations. Empirically, we demonstrate the superiority of FedTrip via extensive evaluations. To achieve the target accuracy, FedTrip outperforms the state-of-the-art baselines in terms of significantly reducing the total overhead of client-server communication and local computation.

Index Terms—Federated Learning, Data Heterogeneity, Resource Efficiency

I. INTRODUCTION

Over the last few decades, massive data have brought about the dramatic development of extensive Artificial Intelligence (AI) applications [1]–[4]. In real life, data are produced by ubiquitous sensing and computing devices, such as mobile phones and wearable devices [5]–[7]. However, in the traditional centralized learning paradigm, raw data are required to be gathered from decentralized devices and transmitted to the central server, which causes unavoidable privacy disclosure and unreasonably high communication overhead.

To alleviate the above issue, Federated Learning (FL) [8]– [10], a distributed learning paradigm that enables participants to collaboratively train a global model without local data exchange, has emerged as an important paradigm and attracted a lot of research interest [11]–[13]. In the fundamental FL algorithm, FedAvg [14], the clients in the FL system train the local models on their private data for multiple local iterations, and upload their updated models to the server for generating an aggregated global model. With no data exchange and periodic model aggregation, FL has significant potential to facilitate AI applications in practice [15].

Nevertheless, FL confronts a key challenge of data heterogeneity [16]–[18], which means that data distributions among clients follow the nonindependent and identically distributed (non-IID) characteristic. This phenomenon inevitably causes apparent update inconsistency among local models [19], [20] and a decline in model generalizability [21]. As a result, the resource consumption for training a model to achieve the desired performance tends to remarkably increase.

To date, various kinds of approaches have been proposed to mitigate the impact of data heterogeneity [22]-[24]. Among them, model regularization [25]–[28] is the general solution, which focuses on constraining the training divergence between the global model and local models by introducing regularization terms into the local loss function. However, constraining model update inherently limits the convergence potential in the local training process [29]. The regularization terms constrain the update divergence but directly prevent the local model from exploring the parameter space far from the global model, where there possibly exists useful information that helps discover the superior local models and promotes model convergence. In addition, the above methods overlook the useful model information that can be learned from historical local models, which is viewed as the diversity of knowledge representation among local models. Consequently, insufficient model information utilization causes slow convergence.

To sufficiently utilize the information from historical local models, a model representation method MOON [30] is proposed. It modifies model updates via designing a loss function based on contrastive learning, whose input terms are the representational outputs of the global model, the current local model, and the historical local model. However, it requires a mass of feedforward operations for extracting feature representation and leads to tremendous computation cost. To date, there is no method that is able to sufficiently utilize the model information with low resource consumption, aiming to settle data heterogeneity.

Motivated by the limitations of existing studies, we propose a novel model regularization method named FedTrip. We expand the triplet loss function [31] to the model level for measuring the divergence of model parameters in the model regularization style. Specifically, a triplet regularization term is added to the local loss function. This term helps the current model to stay close to the global model for guaranteeing update consistency and keep away from historical local models for efficiently exploring parameter spaces. Our proposed FedTrip is able to efficiently extract helpful convergence information during the training process with attaching minor operations, negligible computation cost, and no additional communication cost. Compared to existing methods, FedTrip achieves sufficient model information utilization and realize convergence acceleration with very low resource consumption under data heterogeneity.

The main contributions are summarized as follows:

- In order to overcome the impact of data heterogeneity with low resource consumption, we propose a novel and effective model regularization method in FL under the circumstance of data heterogeneity, named FedTrip. Specifically, we introduce a triplet regularization term into the local loss function. This term decreases the global-local convergence discrepancy and simultaneously increases the current-historical model difference with negligible computation cost and no additional communication cost, which can guarantee consistent model updates and obtain more useful training information.
- In addition, we theoretically analyze the convergence property of FedTrip based on easily-satisfied convergence conditions in the FL system. Theoretical results demonstrate that FedTrip can achieve faster convergence with given hyperparameters than FedProx.
- Extensive experiments are conducted to verify the performance of FedTrip. Experimental results show its superiorities in terms of client-server communication overhead and local computation overhead under various settings and hyperparameters. Especially, FedTrip is satisfactory under strict settings of FL, including highly-skewed data heterogeneity and low client participation ratio.

II. RELATED WORK

In the FL training paradigm, clients are randomly selected to execute local training at each communication round, during which clients' local data are not allowed to share with others. Following [19], we illustrate the impact of data heterogeneity on local model updates as shown in Fig. 1. When local data are IID, for each client k, the local optimum w_k^* is close to the global optimum w^* , and the updates of w_k^t are consistent with other clients. When local data are non-IID, the local optimum w_k^* cannot align with the global optimum w^* , and the updates of w_k^t have inconsistency with others. This inevitably causes the obvious divergence of the local models. Although this issue has attracted many research interests, we only focus on the related methods that intuitively inspire us to tackle the impact of data heterogeneity.

A. Model Regularization

Recent studies have focused on utilizing model regularization to mitigate the impact of data heterogeneity in FL. In these works, the local training objective of clients not only measures the empirical risk over local data but also attaches



Fig. 1. Illustration of model updates in federated learning with IID data and non-IID data settings. Circles are the local model updates and optima. Rectangles are the global model updates and optimum.

additional regular terms to reduce the training divergence among local models. FedProx [25] is the first study that adds a proximal term to effectively limit the local model updates by restricting local models to approximate the global model. Based on FedProx, FedDANE [27] further leverages a gradient correction term to improve training performance. Although FedDANE has more regularization terms and encouraging theoretical guarantee, it consistently underperforms FedProx in evaluations. FedDyn [26] dynamically updates the local loss function by adding a term to guarantee the similarity between local gradients and the parameters, which ensures that the local optima are asymptotically consistent with the global optimum.

However, existing model regularization methods mainly contribute to directly constraining the discrepancy between local models and the global model via adding regularization terms into the loss function, which potentially prevents local models from exploring the superior parameters and obtaining more useful convergence information. Moreover, they overlook the information in historical local model, leading to insufficient information utilization and unsatisfactory model performance.

B. Model Representation

A few recent studies also pay attention to tackling the issue of data heterogeneity in FL using model representation, which devote to optimizing specific loss functions based on modifying feature representations of the local models under the guidance of global model representation. FedGKD [32] aligns the feature representations of the global model and local models via knowledge distillation [33], which achieves relatively consistent local and global representations by guiding local model training through the global model. However, this method still overlooks the information in historical local model. MOON [30] takes historical information into consideration and designs a model-contrastive loss function based on contrastive learning [34] to tune the feature representation similarity among the global model, the current local model, and the historical local model. However, this method may not be practical because it requires $3 \times$ feedforward computation operations for calculating feature representations.

Different from the above methods, we design a novel triplet regularization term inspired by the triplet loss [31], providing the insight of narrowing the convergence divergence between

TABLE I Comparison of existing methods on information utilization and resource cost of method operations.

Methods	Information utilization	Resource cost
Model regularization	Insufficient	Low
Model representation	Sufficient	High
FedTrip	Sufficient	Low

the current local model and the global model to guarantee the consistency training update while putting the current local model away from historical local models for exploring the superior parameters. As shown in Table I, our proposed FedTrip integrates the advantages of model regularization and model representation methods, aiming to achieve sufficient information utilization with negligible computation cost. The quantitive analysis of computation and communication consumption at each communication round of related methods is presented in Appendix A.

III. PROBLEM FORMULATION AND THEORETICAL ASSUMPTIONS

A. Problem Formulation

Without loss of generality, we assume a FL system consisting of N clients and a central server. Let $\mathcal{N} = \{1, 2, \dots, N\}$ denote the set of clients, and the private data that each client $k \in \mathcal{N}$ stores are denoted by \mathcal{D}_k . Considering of data heterogeneity, data distributions across clients differ and follow non-IID in our setting. The goal of our system is to minimize the average loss over heterogeneous data sampled from distributed clients, which is expressed as:

$$\min_{w \in \mathbb{R}^d} f(w) = \min_{w \in \mathbb{R}^d} \frac{1}{N} \sum_{k=1}^N F_k(w; \mathcal{D}_k), \tag{1}$$

where w is the parameters of the global model, and $F_k(w; \mathcal{D}_k)$ measures the local empirical risk on client k over \mathcal{D}_k .

The whole process of FL splits into multiple communication rounds. At the *t*-th round, the server first randomly selects a fixed number K of clients, denoted as S^t , to participate in training. After client selection, the server synchronously transmits the global model w^{t-1} to the selected clients. Afterwards, all clients in S^t perform local model training based on their private data in parallel, and generate the updated local models $\{w_k^t\}$. All updated local models are transmitted back to the server when all clients in S^t finish training. The server then aggregates local models to form the updated global model as

$$w^t = \sum_{k=1}^K a_k^t w_k^t, \tag{2}$$

where a_k^t indicates the weighted coefficient of client k, and $\sum_{k \in S^t} a_k^t = 1$. In the fundamental FL method FedAvg [14], $a_k = \frac{|\mathcal{D}_k|}{|\mathcal{D}_S^t|}$, where $|\mathcal{D}_k|$ is the number of data samples in client k and $|\mathcal{D}_S^t|$ is the total data size of S^t .



(a) Server, 50 rounds (b) Client 1, 50 rounds (c) Client 1, 30 rounds

Fig. 2. T-SNE visualization of the global model at round 50 and of the local model of client 1 at round 30 and 50 on the test dataset.

B. Theoretical Assumptions

For the convenience of theoretical analysis, we give a few standard assumptions (see e.g., [25], [35]), and leverage them to conduct convergence analysis in Section IV-C.

Assumption 1 (*L-smooth*). The stochastic gradient of loss function at each client k is *L* smooth, i.e.,

$$\|\nabla F_k(w_i) - \nabla F_k(w_j)\| < L \|w_i - w_j\| \ \forall w_i, w_j \in \mathbb{R}^d.$$
(3)

Assumption 2 (Bounded Gradient Dissimilarity). The norm of stochastic gradients between the loss function of each client k and the global objective function are bounded, i.e.,

$$\|\nabla F_k(x)\|^2 \le B^2 \|\nabla f(x)\|^2.$$
(4)

IV. METHOD

In this section, we first highlight our motivation of designing FedTrip, and then elaborate on the details of FedTrip. Finally, we conduct the convergence analysis of FedTrip.

A. Motivation

To verify the intuition of our method, we train a CNN model using FedAvg [14] on MNIST [36] (we use the default experiment settings, details of settings and the models can be seen in Section V). Fig. 2 displays the t-SNE [37] visualization of features from the test dataset at the last communication round. The feature representations of all classes in the global model can be distinguished. However, the feature representations of some classes in the local models are still mixed (seen in Figs. 2(b)). This indicates that the performance of the global model is better than that of the local model. Moreover, along the training process, the newer local model tends to outperform the older one (seen in Figs. 2(b), 2(c)). In this view, the performance of the current local model is better than that of historical local models as well.

Depending on this observation, existing model regularization methods have better performance by closing the current local model to the global model via constraining model updates. Nevertheless, the information on historical local models has been overlooked so far in the existed studies. A recent model representation method, MOON [30], takes historical model information into consideration. However, MOON is not resource-efficient because of the tremendous computation cost related to feedforward representation operation. Therefore, a focus on designing a method that concurrently considers localglobal divergence and current-historical correlation with low computation cost should be the primary concern.

B. Method Description

Based on the above analysis and discussion, we provide a novel insight of adding a triplet regularization term into the local loss function of clients, inspired by [31]. The triplet loss is originally designed in [31] to decrease the distance of similar samples and increase that of distinct samples, and we expand it to the model level. Specifically, we inventively propose a triplet-based loss function, which simultaneously constrains the current local model to be close to the global model as well as keeping away from historical local model, with negligible computation cost on attaching operations. The loss function of our FedTrip is defined as:

$$\mathcal{L} = F(w) + \frac{\mu}{2} \left[\|w_{local} - w_{global}\|^2 - \xi \|w_{local} - w_{historical}\|^2 \right].$$
(5)

In equation 5, the first term F(w) represents the original local loss function. The second term $||w_{local} - w_{global}||^2$ intends to guide local models closer to the global model, which keeps the consistency updates among local and global models. The third term $-\|w_{local} - w_{historical}\|^2$ intends to bring current local models away from historical local models, bringing the benefits that the current local model enables to search for the superior parameters and obtains more convergence information. μ and ξ are the hyperparameters to measure the effect of the latter two terms. Note that, the value of \mathcal{E} is set as the interval between the current round and the last round of participating in training.

Algorithm 1 FedTrip

Input: the global round T, the learning rate α , the coefficients μ and ξ

Output: the final model w^T

1: for t = 1 to T do

- The server randomly selects K clients as S^t and deliv-2: ers the global model w^{t-1} to them
- for client in S^t in parallel do 3:
- Let $w_k^t = w^{t-1}$, and load the historical local model 4: \tilde{w}_k^{t-1}
- for batch data ζ_k^t do 5:

6: Calculate local loss
$$F_k(w_k^t; \zeta_k^t)$$

8:
$$w_k^{\iota} = w_k^{\iota} - \alpha \mathcal{U}($$

- end for 9:
- end for 10:
- Clients in \mathcal{S}^t upload w_k^t to server 11:
- The server aggregates the local models via $w^t =$ 12: $\sum_{k \in \mathcal{S}^t} \rho_k w_k^t$ 13: end for
- 14: return w^T

The details are summarized in Algorithm 1. At the beginning of the *t*-th communication round, the server randomly selects K clients named \mathcal{S}^t and delivers the global model w^{t-1} to these clients. We denote w_k^t as the current local model and



Fig. 3. Illustration of model updates of 2 clients with 3 steps in FedProx and FedTrip.

 \tilde{w}_{ι}^{t-1} as the historical model at client k, which is generated at the last local training. After receiving w^{t-1} , each selected client begins its local training (line 5). In line 6, client k trains its local model with mini-batch ζ_k^t , calculates the original loss $F_k(w_k^t)$. Then local gradients are generated by the original loss value and regularization items (line 7). After obtaining the gradients of clients, each selected client updates its local model according to the specific optimization algorithm \mathcal{U} (line 8). The clients in S^t upload local models to the server when all clients finish local training. Finally, the server aggregates the uploaded local models to obtain the updated global model w^t (line 12).

Fig. 3 intuitively depicts the benefits of FedTrip. w_1^* and w_2^* represent the empirical local optima of client 1 and client 2 respectively, and w^* represents the global optimum. The local SGD updates move towards the average of clients' local optimum $\frac{w_1^* + w_2^*}{2}$, which is obviously different from w^* . In the typical model regularization method, FedProx, the local SGD updates of client $k \in \{1, 2\}$ are constrained by $w_k^t - w^t$. However, this method potentially limits the convergence process, as the projection of local gradients towards the direction of the local optimum can be partially counteracted by the gradients generated by the regularization term. Creatively, in FedTrip, the local gradients under the guidance of the historical model overcome the drawback of FedProx, and the local model has the potential to explore superior parameters with the guarantee of update consistency. Therefore, our proposed FedTrip has the advantage of absorbing more useful information, which enables the local model to move towards the global optimum w^* quickly, thus accelerating training convergence.

C. Convergence Analysis

We theoretically analyze the global model convergence of FedTrip. This analysis mainly refers to FedProx [25] and FedDANE [27]. Firstly, we define a parameter γ to formulate the inexactness of local optimization.

Definition 1 (γ -inexact optimization). Let w_k^{t+1} denote the updated local model of client k based on local optimization at the t + 1-th round, and it satisfies $\|\nabla h(w_k^{t+1}; w^t)\| < \gamma \|\nabla F_k(w^t)\|$ with $\gamma \in [0, 1)$, where $\nabla h(w_k^{t+1}; w^t) = \nabla F_k(w_k^{t+1}) + \mu \left(w_k^{t+1} - w^t\right) - \xi (w_k^{t+1} - \tilde{w}_k^t) \right)$. **Theorem 1**. Assume that the functions F_k are convex, h_k is μ -strongly convex. Given by assumptions mentioned in Section III-B, we have the expected decrease in the global objective function as:

$$\mathbb{E}_{S_t}[f(w^{t+1})] \le f(w^t) - \rho \|\nabla f(w^t)\|^2 - Q^t, \qquad (6)$$
$$\rho = \left(\frac{1 - \gamma B}{\mu} - \frac{L(1 + \gamma)B}{\mu^2} - \frac{L(1 + \gamma)^2 B^2}{2\mu^2}\right).$$

where Q^t is the extra items generated by the historical information item. The expectation of coefficient of Q^t is proportional to the participation ratio of a client at each round p. Let $\gamma = 0$, which means $F_k(w)$ has the exact answer, then

$$\rho = \frac{1}{\mu} - \frac{LB}{\mu^2} - \frac{LB^2}{2\mu^2}.$$

If μ, L, γ, ξ satisfy, we have $\rho > 0$ and $Q^t > 0$, and the local objective function has the expected decrease as:

$$\mathbb{E}_{S_t}[f(w^{t+1})] \le f(w^t) - \rho \|\nabla f(w^t)\|^2 - Q^t.$$
(7)

Note that the value of ρ in FedTrip is equal to ρ in Fedprox, we can get the identical decrease proportional to $\|\nabla f(w^t)\|^2$ with FedProx. Besides, Q^t makes f(w) have the faster convergence rate of FedTrip than that of FedProx with the help of historical model information. The main coefficient in Q^t is $E_k[\xi_k^t] = \frac{p \ln p}{p-1}, t \to +\infty$, where p is the client participation rate. As $E_k[\xi_k^t]$ is monotonically increasing, a low p demonstrates a slow convergence rate. The detailed convergence analysis is referred to the Appendix B. In summary, FedTrip is a resource-efficient and fast convergence method with given hyperparameters.

V. EXPERIMENTS

A. Experimental Settings

We investigate our method and comparable baselines on an open-source federated learning framework Plato¹ with PyTorch [38] 1.9.1 backend, whose data partition method is based on LEAF [39]. Our experiments are executed on a workstation with an Intel Xeon Gold 5218 CPU @ 2.30GHz, a RAM of 376 GB, and one Nvidia GeForce 3090 GPU.

Datasets: We employ MNIST [36], FashionMNIST (FM-NIST) [40], EMNIST [41], and CIFAR-10 [42] for image classification task. The datasets cover different attributes, dimensions, and numbers of categories, which are listed in Table II. Thereinto, 1 channel and 3 channels indicate grayscale and RGB images, respectively. Client samples indicate the number of data samples at each client.

Models: We train a MultiLayer Perceptron (MLP) on MNIST and FMNIST datasets. MLP consists of 2 fully connected layers with 100 and 10 neurons. The first fully connected layer is followed by ReLU activation [43]. A simple Convolution Neural Network (CNN) is used for training on MNIST, FMNIST, and EMNIST. The CNN is modified based on LeNet5 [36], consisting of 3 convolutional layers with 5×5 filters followed by two fully connected layers with 84 and 10

 TABLE II

 Description of Datasets in the experiment.

Datacat	Total	Classos	Channels	Client
Dataset	Samples	Classes	Channels	Samples
MNIST	60,000	10	1	600
FMNIST	60,000	10	1	1,000
EMNIST	112,800	47	1	3,000
CIFAR-10	50,000	10	3	2,000

 TABLE III

 COMMUNICATION AND COMPUTATION STATISTICS OF MODELS

Model	Communication(MB)	Params(M)	MFLOPs
MLP	0.3	0.8	0.08
CNN	0.24	0.62	0.42
AlexNet	10.42	2.72	145.93



Fig. 4. The label distributions at clients in four settings of data heterogeneity.

neurons. Moreover, AlexNet [44] is trained on CIFAR-10.

Data Partitioning: We adopt two popular non-IID data partitioning ways: Dirichlet distribution and orthogonal distribution. We generate two types of data heterogeneity via the following Dirichlet distribution to sample data labels. First, each client draws a probability vector using the Dirichlet distribution with a concentration parameter α , which corresponds to the prior data distribution of each class. The probability vectors are generated based on different random seeds and are used to sample data without replacement for clients. The sampling process does not stop until the number of data samples is assigned to the preset partition number. In our experiments, we implement 2 types of Dirichlet distributions with $\alpha = 0.1, 0.5$, named Dir - 0.1 and Dir - 0.5.

Moreover, we simulate orthogonal data distribution, where clients are partitioned into multiple clusters. For each cluster, the data samples owned by inner clients have non-overlapped classes with those of other clusters, and the data samples of clients in each cluster are IID sampled. Concretely, we set two types of orthogonal data distribution by dividing clients

¹https://github.com/TL-System/plato

 TABLE IV

 Comparison of communication rounds until the global model achieves the target accuracy.

Methods	MLP			CN	CNN					Alex	Net						
MNIST		Г-87%	7% FMNIST-75%		MN	MNIST-90%		FMNIST-75%		EMNIST-62%			CIFAR-50%				
FedTrip	28			9		24			19			32			46		
FedAvg	49 🔳		$1.75 \times$	19	$2.11 \times$	39		$1.63 \times$	52		$2.73 \times$	45		$1.4 \times$	74		1.61×
FedProx	53		$1.89 \times$	16	$1.78 \times$	41		$1.71 \times$	45		$2.37 \times$	45		$1.4 \times$	75		1.63×
SlowMo	46		$1.64 \times$	26	2.89×	40		$1.67 \times$	65		3.42×	92		$2.88 \times$	87		1.89×
MOON	25		$0.89 \times$	14	$1.56 \times$	46		1.92×	35		$1.84 \times$	44		$1.38 \times$	84		1.75×
FedDyn	28		$1 \times$	17	$1.89 \times$	40		$2.08 \times$	51		$2.68 \times$	97		3.03×	79		1.72×

into 5 and 10 clusters in our experiments, which are named Orthogonal - 5 and Orthogonal - 10 respectively.

Fig. 4 shows the local data distributions of clients on MNIST dataset in 4 heterogeneity types. The majority of clients contain mostly 3 or 4 classes of data samples under Dir - 0.5, and 1 or 2 classes of data samples under Dir - 0.5. Under Orthogonal - 5 and Orthogonal - 10, each client only has 2 and 1 classes of data samples. For example, Client 1 only have data samples with classes 0, 1 and class 0.

Baselines: We compare the convergence performance of our proposed FedTrip with FedAvg [14], FedProx [25], SlowMo [45], MOON [30] and FedDyn [26]. The default local optimizer is SGD with momentum (SGDm) [46], a fixed learning rate of 0.01 and the momentum coefficient of 0.9. Considering SGDm may results in performance degradation in some circustances, SlowMo and FedDyn choose SGD as the training optimizer. The hyperparameters of these methods are: $\mu = 1.0$ for all MLP experiments and $\mu = 0.4$ for others in FedTrip, $\mu = 0.1$ in FedProx, $\alpha = 1$ for the expertiments on MNIST dataset and $\alpha = 0.1$ for other datasets in FedDyn, $\mu = 1, \tau = 0.5$ in MOON.

The default number of communication rounds, batch size, and local epoch are set as 100, 50, and 1, respectively. Besides, the server randomly selects 4 devices from 10 devices. We perform the 10-trial repeating experiment and report the average convergence performance.

B. Resource Efficiency

We verify the effectiveness of resource efficiency of FedTrip from the perspective of client-server communication and local computation. The results show that FedTrip is able to save resources significantly.

Communication Efficiency: As all aforementioned methods have exactly the same amount of communication volume per communication round, the total amount of communication bits is proportional to the number of communication rounds. We define the number of communication rounds at which the global model achieves the target accuracy as the evaluation metric. Table IV shows the results of these methods on MLP, CNN, and AlexNet models under Dir - 0.5. The dark grey bars denote the number of communication rounds to achieve the target accuracy of the global model using different methods. Thereinto, the longest dark grey bar indicates that the corresponding method has the maximum number of

TABLE V GFLOPS AMONG METHODS DURING THE TRAINING PROCESS

Model	Case	FedTrip	FedAvg	FedProx	SlowMo	MOON	FedDyn
MLP	MNIST	1.441	2.334	2.626	2.191	3.573	1.441
	FMNIST	0.772	1.509	1.321	2.064	3.335	1.458
CNN	MNIST	6.161	9.897	10.465	10.151	35.02	10.269
	FMNIST	8.13	21.993	19.144	27.491	44.409	21,822
	EMNIST	41.077	57.097	57.431	116.733	167.486	124.513
AlexNet	CIFAR	13,446	21,596	21,906	25,392	73,549	23,091

communication rounds. Besides, the difference in the number of communication rounds between our proposed FedTrip and other methods can be shown by the blue lines.

Among all the methods, FedTrip and MOON are the fastest. This demonstrates that absorbing information from both the global model and historical local models can effectively accelerate model convergence. Compared to MOON, FedTrip further reduces the communication rounds by 31.63%, which shows that FedTrip absorbs model information more efficiently so as to further improve the convergence rate. Compared to the fundamental method FedAvg, FedTrip is $1.4-2.73 \times$ faster to achieve the target accuracy on training models, and the amount of communication rounds of FedTrip reduces by 44.02% on average. We conclude that FedTrip shows the best performance on reducing communication overhead.

Local Computation Efficiency: Based on our theoretical analysis of computation cost over attaching operations in these methods (see Appendix A), the computation cost of MOON is $50\times$, $171.4\times$ and $1,336\times$ as much as that of FedTrip at each local iteration on training MLP, CNN and AlexNet, respectively. We utilize the total GFLOPs of feedforward and attaching operations in these methods to measure computational efficiency, which are listed in Table V.

From Table V, it can be seen that FedTrip reduces the computation cost by 39.58% on average, compared to the baseline method with the least GFLOPs in each experiment case. The local computation overhead of MOON is $4.52\times$ that of FedTrip, which demonstrates that FedTrip can obtain more convergence information with much less computation cost. As MOON simultaneously obtains the information of the global and historical models via multiple feedforward operations, it is the most computation-inefficient method. Compared to the fundamental method FedAvg, FedTrip reduces the local



Fig. 5. The convergence curves of CNN with 2 types of data heterogeneity on 3 datasets.



Fig. 6. Boxplots of test accuracy of CNN and MLP with 4 types of data heterogeneity on FMNIST dataset. The accuracy of MOON under Orthogonal - 10 is significantly lower than others, so it is invisible in (a).

computation overhead by up to 42.27%.

C. Data Heterogeneity

Fig. 5 illustrates the test accuracy at each communication round for training CNN on different datasets among above methods. All the curves are smoothed by the exponential moving average. As the value of μ is small in FedProx, the convergence performance of FedProx is generally close to that of FedAvg. As the regularization term in the local loss functions constrains the divergence of update directions among clients, FedProx becomes effective under orthogonal heterogeneity (see in Figs. 5(e), 5(f)), where the local updates diverge considerably. FedDyn and SlowMo underperform other methods on EMNIST dataset. MOON [30] outperforms other baseline methods in Dir - 0.5, which indicates the advantages of absorbing the information of global and historical models. FedTrip is competitive to data heterogeneity as it outperforms other baselines in most of experiment cases.

Fig. 6 illustrates the final accuracy, the average accuracy of the global model over the last 10 communication rounds of CNN and MLP on FMNIST dataset among all methods. FedTrip performs the highest final accuracy in most of the experiments under various heterogeneity types. FedDyn has the highest accuracy on MLP under the orthogonal heterogeneity types, which shows its superiority of the related regularization term for aligning the local model and local gradients. Compared to the experimental results under the Dirichlet distribution, the convergence improvement of FedTrip is more remarkable than the improvement in the experiments under the orthogonal distribution. Although MOON can simultaneously obtain the global model and historical model information, it

TABLE VI THE NUMBER OF COMMUNICATION ROUNDS OF CNN TO ACHIEVE THE TARGET ACCURACY IN 4-50.

		MNI	FMNIST				
Method	Dir-0.1	Dir-0.5	Orthogonal-5	Dir-0.1	Dir-0.5	Orthogonal-5	
	87%	90%	85%	65%	75%	60%	
FedTrip	30	19	43	19	15	35	
FedAvg	$1.6 \times$	$1.74 \times$	$2.14 \times$	$2.74 \times$	$3 \times$	$2.51 \times$	
FedProx	$1.8 \times$	$1.71 \times$	$1.7 \times$	$2.68 \times$	$2.87 \times$	$2.14 \times$	
SlowMo	$1.87 \times$	$1.71 \times$	$1.7 \times$	$4.21 \times$	$4.67 \times$	$>2.86\times$	
MOON	$2.33 \times$	$1.32 \times$	$2.28 \times$	$4 \times$	$2.67 \times$	$>2.86\times$	
FedDyn	$2.17 \times$	$3 \times$	2.28 imes	$4.16 \times$	$5.07 \times$	$>2.86\times$	

conducts the worst performance in Orthogonal - 10. This reveals that this model representation method is not suitable for all data distributions, especially highly-skewed data distributions. Overall, FedTrip achieves $2.53 \times$ and $1.38 \times$ convergence acceleration compared to the state-of-art method MOON in Dir - 0.1 and Orthogonal - 10 respectively. We attribute this to that FedTrip ables to mitigate the server fluctuation in the convergence trajectory in heavily-skewed data distributions.

D. Scalability

We discuss the scalability of FedTrip based on the client participation type that the server randomly selects 4 devices from 50 devices. The convergence performances across different models and data heterogeneity types are listed in Table VI. Symbol > in Table VI indicates that the global model of the specific method does not achieve the target accuracy at the last communication round.

The communication rounds of these methods to achieve the target accuracy in 4-50 are less than that in 4-10 with the same hyperparameters, which is benefit from the larger number of total data samples. With the details of FedTrip in Section IV, to scale the influence of the historical local model, ξ in FedTrip is scaled by the gap between the current round and the last participated round. The expectation value of ξ decreases to $\frac{1}{5}$ of ξ in 4-10. Among experiments, FedTrip performs the fastest convergence in 4-50. Compared to FedAvg and MOON, FedTrip reduces communication rounds by up to 56.1% and 54.82% respectively. The performance of MOON degrades in this setting, which shows the limitation of MOON in low client participation environments. In summary, FedTrip consistently yields substantial resource savings compared to baselines across various client participation settings.

E. Influence of Aggregation Intervals

In this part, we enlarge the number of local training epochs to 5 and 10 at each communication round. The experiments run under settings with Dir - 0.5 and 4-10. We set $\mu = 0.4$ in FedTrip. We list the test accuracy of each method at the 10-th and 20-th communication round in Table VII. FedTrip consistently achieves the highest accuracy among different aggregation intervals. With the increase of local training iterations per communication round, the average accuracy of

TABLE VII The accuracy among methods with 5 and 10 of local epochs.

# Local Epochs	#Rounds	FedTrip	FedAvg	FedProx	SlowMo	MOON	FedDyn
5	10	96.36	95.49	93.08	84.55	95.26	87.93
	20	97.18	96.71	95.95	92.88	96.88	93.49
10	10	97.49	97.38	95.84	87.79	96.99	93.11
	20	97.95	97.84	97.25	95.15	97.84	95.93

all methods at each round is improved. Although a large aggregation interval exacerbates the staleness of historical local models, our method can still obtain the effective information from the historical models to accelerate convergence. SlowMo and FedDyn have unsatisfactory performance owing to the frequency reduction of the additional operations at the server, resulting in incorrect updates.

F. Sensitivity Analysis of μ in FedTrip

To explore the influence of μ on the convergence, we compare the model accuracy and convergence rate of FedTrip by varying μ from 0.1 to 2.5. Note that the final accuracy is defined as the highest test accuracy in the training process, which indicates the best performance across different values of μ . The model, dataset, and participation type are CNN, MNIST, and 4-10 respectively. The results are shown in Fig. 7. The blue and the orange circles represent the final accuracy and the number of communication rounds required to achieve the 90% test accuracy of the global model. Note that, the radii of circles represent the variance of corresponding metrics.

Under all settings, FedTrip eventually converges successfully. As shown in Fig. 7, FedTrip suffers a lower convergence rate when the value of μ is small. It accelerates convergence and improves test accuracy to 93.48% and 94.06% under Dir - 0.1 and Dir - 0.5 when $\mu = 0.4$. Afterward, the convergence is still accelerated at the sacrifice of accuracy degradation when the value of μ increases to approximately 1.5. With further increasing μ , the number of communication rounds increases, and the final accuracy decreases. Under Dir - 0.1, the test accuracy of FedTrip fluctuates more considerably and degrades faster with increasing μ than that under Dir - 0.5. Under Orthogonal setting, our method has the more stable performance than that under Dirichlet data heterogeneity with the change of μ . The performance fluctuates dramatically When $\mu > 2$, but the test accuracy drops to 80% only when $\mu = 2.5$. As shown in Figure 7 (d), the performance is sensitive to μ . With the value of μ increases, the test performance considerably degrades.

Consequently, for devices with limited resource budgets or with less-strict performance requirements, a large μ is a better option. Conversely, we need to set a small μ for the cases with high performance requirements.

VI. CONCLUSION

In this paper, we propose a resource-efficient FL method named FedTrip based on the insight of constraining local



Fig. 7. The performance influence of μ on CNN model and MNIST and MLP model on FMNIST.

model close to the global model while keeping away from the historical model, aiming to mitigate the impact of inconsistency model update derived from data heterogeneity and effectively obtain the information that helps fast convergence. Specifically, we add a triplet regularization term to the clientside loss function to absorb the useful information from historical local model with negligible computation cost. Our experiments show that FedTrip outperforms the state-of-the-art methods in terms of reducing computation and communication overhead under the circumstances of varying data heterogeneity, client participation and hyperparameter settings. In future,

TABLE VIII Comparison of FedTrip with related state-of-the-arts on the computation and communication overhead.

	Computation	Communication
Method	Overhead (FLOPs)	Overhead
SCAFFOLD [17]	2(K+1) w + n(FP + BP)	2 w
MimeLite [35]	n(FP + BP)	2 w
MOON [30]	K(M(1+p)FP)	0
FedProx [25]	2K w	0
FedDyn [26]	4K w	0
FedTrip	4K w	0

we will further discuss the influence of ξ and analyze the convergence rate of FedTrip in general convex and non-convex cases.

APPENDIX A DISCUSSION ON RESOURCE CONSUMPTION OF ATTACHING OPERATIONS IN METHODS

This part illustrates the computation and communication consumption of FedTrip compared with related state-of-theart methods. FedAvg is the baseline, and all methods use the SGDm optimizer. Table VIII displays the consumption at each communication round of attaching operations in these methods. We define K, M, n, |w| as the number of local iterations, batch size, local data samples, and the size of model parameters. In addition, \mathcal{FP} and \mathcal{BP} represent the computation overhead of feedforward and backpropagation for a single data sample, respectively.

SCAFFOLD and MimeLite require extra computation of full-batch gradients to estimate the true gradients, which requires $n(\mathcal{FP} + BP)$ computation overhead. Note that the computation overhead of \mathcal{FP} and \mathcal{BP} is much larger than |w|. For example, the computation overhead of \mathcal{FP} is 342 × as much as |w| on CNN model. In addition, they require extra transmission in both downstream and upstream communication. The size of transmission is 2|w|. MOON requires extra $(1+p)\mathcal{FP}$ for each local iteration, where p denotes the number of history models used in the local iterations. FedTrip only requires 4|w| computation overhead at each local iteration, much smaller than that of MOON.

In general, although existing methods can alleviate data heterogeneity and improve convergence performance, they require a large amount of local computation or client-server interactions because of their attaching operations. The attaching operations of FedTrip not only doesn't increase additional communication overhead, but its computation overhead is much smaller than that of other methods, almost negligible.

APPENDIX B Sketch of convergence proof

We prove the convergence analysis of FedTrip by referring to the proof of FedProx [25] and FedDANE [27]. Besides, one lemma is adapted from Scaffold [17], which we will apply in $\langle \nabla f(w^t), w^{t+1} - w^t \rangle$:

$$\langle \nabla f(x), y - z \rangle \le f(y) - f(z) - \frac{\mu}{4} \|y - z\|^2 + L \|z - x\|^2.$$
(8)

First, we define e_k^t such that:

$$\nabla F_k(w_k^{t+1}) + \mu(w_k^{t+1} - w^t) + \mu \xi_k^t(\tilde{w}_k^t - w_k^{t+1}) = e_k^t$$
$$\|e_k^t\| \le \gamma \|\nabla F_k(w^t)\|,$$

and we define $\mathbb{E}_k[w_k^{t+1}] = \bar{w}^{t+1}$. We have

$$(\mathbb{E}_{k}[w_{k}^{t+1}] - w^{t}) + \mathbb{E}_{k}[\xi_{k}^{t}(\tilde{w}_{k}^{t} - w_{k}^{t+1})] = -\frac{1}{\mu}\mathbb{E}_{k}[\nabla F_{k}(w_{k}^{t+1})] + \frac{1}{\mu}\mathbb{E}_{k}[e_{k}^{t}].$$
(9)

Then we set $\hat{w}_k^{t+1} = \arg \min_w h_k(w; w^t)$. It's obviously that $\nabla h_k(\hat{w}_k^{t+1}, w^t) = 0$. Due to the μ -strongly convex of h_k , we can get

$$\begin{aligned} \|\hat{w}_{k}^{t+1} - w_{k}^{t+1}\| &= \frac{1}{\mu} \|\nabla h(\hat{w}_{k}^{t+1}, w^{t}) - \nabla h(w_{k}^{t+1}, w^{t})\| \\ &= \frac{1}{\mu} \|e_{k}^{t}\| \leq \frac{\gamma}{\mu} \|\nabla F_{k}(w_{k}^{t})\|, \\ \|\hat{w}_{k}^{t+1} - w^{t}\| &= \frac{1}{\mu} \|\nabla h(\hat{w}_{k}^{t+1}, w^{t}) - \nabla h(w^{t}, w^{t})\| \\ &= \frac{1}{\mu} \|-\nabla F_{k}(w^{t}) - \mu \xi_{k}^{t}(\tilde{w}_{k}^{t} - w^{t})\|, \\ \|w_{k}^{t+1} - w^{t}\| \leq \frac{1+\gamma}{\mu} \|\nabla F_{k}(w^{t})\| + \xi_{k}^{t} \|\tilde{w}_{k}^{t} - w^{t}\|. \end{aligned}$$

$$(10)$$

As $\mathbb{E}_k[\|w_k^{t+1} - w^t\|] = \|\bar{w}^{t+1} - w^t\|$, we use Assumption 1 to get

$$\|\bar{w}^{t+1} - w^t\| \le \frac{B(1+\gamma)}{\mu} \|\nabla f(w^t)\| + \mathbb{E}_k[\xi_k^t \|w^t - \tilde{w}_k^t\|].$$

Here, we further define $\|\tilde{w}^t - w^t\|$ as

$$\begin{aligned} \|\tilde{w}_{k}^{t} - w^{t}\| &\leq \frac{1}{\mu} \|\nabla h(\tilde{w}_{k}^{t}, w^{t}) - \nabla h(w^{t}, w^{t})\| \\ &\leq \frac{1}{\mu} \|\nabla F_{k}(\tilde{w}_{k}^{t}) + \mu(\tilde{w}_{k}^{t} - w^{t}) \\ &- \nabla f(w^{t}) - \xi_{k}^{t} \mu(\tilde{w}_{k}^{t} - w^{t})\| \\ &\leq \frac{1}{(1 - (1 - \xi_{k}^{t}))\mu} \|\nabla F_{k}(\tilde{w}_{k}^{t}) - \nabla F_{k}(w^{t})\| \\ &\leq \frac{1}{\xi_{k}^{t}\mu} \|\nabla F_{k}(\tilde{w}_{k}^{t}) - \nabla F_{k}(w^{t})\|. \end{aligned}$$
(11)

According to L-smooth, we get the recursive inequality of $\|\tilde{w}^t - w^t\|$ and $\|\nabla F_k(\tilde{w}^t) - \nabla F_k(w^t)\|$ as follows:

$$\|\nabla F_k(\tilde{w}^t) - \nabla F_k(w^t)\| \le L \|\tilde{w}^t - w^t\|, \qquad (12)$$

$$\xi^{t} \|\tilde{w}^{t} - w^{t}\| \le \frac{B}{\mu} \|\nabla f(\tilde{w}^{t}) - \nabla f(w^{t})\|.$$
(13)

Then, we get the inequality related to $f(\bar{w}^{t+1})$ and $f(w^t)$ based on L-smoothness, which can be given by:

$$f(w^{t+1}) \leq f(w^{t}) + \langle \nabla f(w^{t}), \bar{w}^{t+1} - w^{t} \rangle + \frac{L}{2} \| \bar{w}^{t+1} - w^{t} \|^{2}$$

$$\leq f(w^{t}) - \frac{1}{\mu} \| \nabla f(w^{t}) \|^{2} + \frac{\gamma B}{\mu} \mathbb{E}_{k} [\langle \nabla f(w^{t}), e_{k}^{t} \rangle]$$

$$- \frac{1}{\mu} \langle \nabla f(w^{t}), \mathbb{E}_{k} [\nabla F_{k}(w^{t+1}) - \nabla F_{k}(w^{t})] \rangle$$

$$+ \mathbb{E}_{k} [\xi_{k}^{t} \langle \nabla f(w^{t}), \tilde{w}_{k}^{t} - w^{t+1} \rangle] + \frac{L}{2} \| w^{t+1} - w^{t} \|^{2}$$
(14)

According to (8) in FedProx, the first three items are identical. So we only need to compare the last two items in (14) to $\frac{L}{2} || w^{t+1} - w^t ||^2$ at (8) in FedProx, which is equal to $\frac{L}{2} || \frac{1+\gamma}{\mu} \nabla F_k(w^t) ||^2$. Firstly, we analyze $\mathbb{E}_k \left[\xi_k^t \langle \nabla f(w^t), \tilde{w}_k^t - w^{t+1} \rangle \right]$.

$$\mathbb{E}_{k}\left[\xi_{k}^{t}\langle\nabla f(w^{t}), \tilde{w}_{k}^{t}-w^{t+1}\rangle\right] = \mathbb{E}_{k}\left[\xi_{k}^{t}\langle\nabla f(w^{t}), \tilde{w}_{k}^{t}-w^{t}\rangle\right] \\ + \mathbb{E}_{k}\left[\xi_{k}^{t}\langle\nabla f(w^{t}), w^{t}-w^{t+1}\rangle\right]$$

$$\mathbb{E}_{k} \left[\langle \nabla f(w^{t}), \tilde{w}_{k}^{t} - w^{t} \rangle \right] \leq \mathbb{E}_{k} \left[F_{k}(\tilde{w}_{k}^{t}) - F_{k}(w^{t}) - \frac{\mu}{4} \| \tilde{w}_{k}^{t} - w^{t} \|^{2} \right]$$
(15)
$$\mathbb{E}_{k} \left[\langle \nabla f(w^{t}), w^{t} - w^{t+1} \rangle \right] \leq \mathbb{E}_{k} \left[F_{k}(w^{t}) - F_{k}(w^{t+1}) - \frac{\mu}{4} \| w^{t} - w^{t+1} \|^{2} + L \| w^{t} - w^{t+1} \|^{2} \right].$$
(16)

 $\frac{L}{2} \| w^{t+1} - w^t \|^2$ in FedTrip is

$$\frac{L}{2} \|w^{t+1} - w^t\|^2 \leq L \left[\|\frac{1+\gamma}{\mu} \nabla F_k(w^t)\|^2 + \mathbb{E}_k \|\xi_k^t(\tilde{w}_k^t - w^t)\|^2 \right]. \quad (17)$$

Combine (15), (16) and (17), we have the additional items Q^t in FedTrip:

$$Q^{t} = \frac{L}{2} \| \frac{1+\gamma}{\mu} \nabla F_{k}(w^{t}) \|^{2} + L \| \xi_{k}^{t} (\tilde{w}_{k}^{t} - w^{t}) \|^{2} + \mathbb{E}_{k} \xi_{k}^{t} \left[F_{k} (\tilde{w}_{k}^{t}) - F_{k}(w^{t+1}) - \frac{\mu}{4} \| \tilde{w}_{k}^{t} - w^{t} \|^{2} - \frac{\mu}{4} \| w^{t} - w^{t+1} \|^{2} + L \| w^{t} - w^{t+1} \|^{2} \right].$$
(18)

Now we analyze the items one by one. In FedProx, μ is set as $6LB^2$ as example, where $B \gg 1$. We adapt it and easily get $\left(\frac{\mu}{4} - L\right) \gg 0$ and $\left(\frac{\xi^t \mu}{4} - L\right) \gg 0$, where $\xi^t = \mathbb{E}[\xi_k^t] \in (0, 1]$. Then we consider $\frac{L}{2} \| \frac{1+\gamma}{\mu} \nabla F_k(w^t) \|^2$.

$$\mathbb{E}_{k}\left[\frac{L}{2}\|\frac{1+\gamma}{\mu}\nabla F_{k}(w^{t})\|^{2}\right] = \frac{LB^{2}(1+\gamma)^{2}}{2\mu^{2}}\|\nabla f(w^{t})\|^{2}.$$
(19)

From (11), we have $-\frac{1+\gamma}{\mu} \|\nabla F_k(w^t)\| \leq \xi_k^t \|\tilde{w}_k^t - w^t\| - [11]$ H. Wang, M. Yurochkin, Y. Sun *et al.*, "Federated learning with matched $||w^{t+1} - w^t||.$

$$\begin{split} \frac{LB^2(1+\gamma)^2}{\mu^2} \|\nabla f(w^t)\|^2 &\leq L\left((\xi_k^t)^2 \|\tilde{w}_k^t - w^t\|^2 \\ &\quad -2\xi_k^t \|\tilde{w}_k^t - w^t\| \|w^{t+1} - w^t\| \\ &\quad + \|w^{t+1} - w^t\|^2\right) \\ &\ll \frac{\mu\xi_k^t}{4} \left(\|\tilde{w}_k^t - w^t\|^2 \\ &\quad + \|w^{t+1} - w^t\|^2\right) \end{split}$$

Lastly, we define L_0 as the local Lipschitz continuity constant of function f and we have

$$||f(\tilde{w}_{k}^{t}) - f(w^{t})|| \le L_{0} ||\tilde{w}_{k}^{t} - w^{t}||$$

 $\begin{array}{l} L_0 \| \tilde{w}_k^t - w^t \| - \frac{\mu}{4} \| \tilde{w}_k^t - w^t \|^2 < 0 \mbox{ is satisfied when } L_0 < \\ \frac{\mu}{4} \| \tilde{w}_k^t - w^{t+1} \| \, \forall t. \end{array}$

If μ, L, γ, ξ stasify, we have $\rho > 0$ and $Q^t > 0$. Assume that $E_{S_t}[f(w^{t+1})] = f(w^{t+1})$, (14) can be written as

$$E_{S_t}[f(w^{t+1})] \leq f(w^t) + \left(\frac{1-\gamma B}{\mu} - \frac{L(1+\gamma)B}{\mu^2} - \frac{L(1+\gamma)^2 B^2}{2\mu^2}\right) \|\nabla f(w^t)\|^2 - Q^t.$$
(20)

If we define that $\gamma = 0$, the inequality is transformed to

$$f(w^{t+1}) \leq f(w^t) - \left(\frac{1}{\mu} - \frac{LB}{\mu^2} - \frac{LB^2}{2\mu^2}\right) \|\nabla f(w^t)\|^2 - Q^t$$
$$\leq f(w^t) - \rho \|\nabla f(w^t)\|^2 - Q^t.$$
(21)

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 62072436) and the National Key Research and Development Program of China(2021YFB2900102).

REFERENCES

- [1] K. He, X. Zhang, S. Ren et al., "Deep residual learning for image recognition," in CVPR, 2016, pp. 770-778.
- A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," [2] NIPS, vol. 30, 2017.
- [3] J. Devlin, M.-W. Chang, K. Lee et al., "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
- [4] Z. Liu, Y. Lin, Y. Cao et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in CVPR, 2021, pp. 10012-10022.
- S. Seneviratne, Y. Hu, T. Nguyen et al., "A survey of wearable devices [5] and challenges," IEEE Commun. Surv. Tutor., vol. 19, no. 4, pp. 2573-2620, 2017.
- [6] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," IEEE Commun. Surv. Tutor., vol. 21, no. 3, pp. 2224–2287, 2019.
- W. Y. B. Lim, N. C. Luong, D. T. Hoang et al., "Federated learning in [7] mobile edge networks: A comprehensive survey," IEEE Commun. Surv. Tutor., vol. 22, no. 3, pp. 2031-2063, 2020.
- J. Konečný, H. B. McMahan, F. X. Yu et al., "Federated learning: Strategies for improving communication efficiency," arXiv preprint arXiv:1610.05492, 2016.
- K. Bonawitz, H. Eichner, W. Grieskamp et al., "Towards federated [9] learning at scale: System design," in MLSys, vol. 1, 2019, pp. 374-388.
- [10] Q. Yang, Y. Liu, Y. Cheng et al., "Federated learning," Synth. Lect. Artif. Intell. Mach. Learn., vol. 13, no. 3, pp. 1-207, 2019.

- averaging," in ICLR, 2020.
- [12] J. Wolfrath, N. Sreekumar, D. Kumar et al., "Haccs: Heterogeneityaware clustered client selection for accelerated federated learning," in IPDPS, 2022.
- [13] Q. Li, Y. Diao, Q. Chen et al., "Federated learning on non-iid data silos: An experimental study," in ICDE, 2022, pp. 965-978.
- [14] H. B. McMahan, E. Moore, D. Ramage et al., "Communication-efficient learning of deep networks from decentralized data," in AISTATS, 2017, pp. 1273-1282.
- [15] G. A. Kaissis, M. R. Makowski, D. Rückert et al., "Secure, privacypreserving and federated machine learning in medical imaging," Nat. Mach. Intell., vol. 2, no. 6, pp. 305-311, 2020.
- [16] A. Khaled, K. Mishchenko, and P. Richtárik, "Tighter theory for local sgd on identical and heterogeneous data," in AISTATS, 2020, pp. 4519-4529.
- [17] S. P. Karimireddy, S. Kale, M. Mohri et al., "Scaffold: Stochastic controlled averaging for federated learning," in ICML, 2020, pp. 5132-5143.
- [18] T. Li, A. K. Sahu, A. Talwalkar et al., "Federated learning: Challenges, methods, and future directions," IEEE Signal Process Mag, vol. 37, no. 3, pp. 50-60, 2020.
- [19] Y. Zhao, M. Li, L. Lai et al., "Federated learning with non-iid data," arXiv preprint arXiv:1806.00582, 2018.
- [20] K. Hsieh, A. Phanishavee, O. Mutlu, and P. Gibbons, "The non-iid data quagmire of decentralized machine learning," in ICML, 2020, pp. 4387-4398.
- [21] P. Kairouz, H. B. McMahan, B. Avent et al., "Advances and open problems in federated learning," arXiv preprint arXiv:1912.04977, 2019.
- [22] J. Wang, Q. Liu, H. Liang et al., "Tackling the objective inconsistency problem in heterogeneous federated optimization," NIPS, vol. 33, pp. 7611–7623, 2020.
- [23] S. Reddi, Z. Charles, M. Zaheer et al., "Adaptive federated optimization," arXiv preprint arXiv:2003.00295, 2020.
- [24] X. Li, M. Jiang, X. Zhang et al., "Fedbn: Federated learning on non-iid features via local batch normalization," arXiv preprint arXiv:2102.07623, 2021.
- [25] T. Li, A. K. Sahu, M. Zaheer et al., "Federated optimization in heterogeneous networks," in MLSys, vol. 2, 2020, pp. 429-450.
- [26] D. A. E. Acar, Y. Zhao, R. Matas et al., "Federated learning based on dynamic regularization," in ICLR, 2021.
- [27] T. Li, A. K. Sahu, M. Zaheer et al., "Feddane: A federated newton-type method," in ACSSC, 2019, pp. 1227-1231.
- [28] G. Kim, J. Kim, and B. Han, "Communication-efficient federated learning with acceleration of global momentum," arXiv preprint arXiv:2201.03172, 2022.
- [29] M. Mendieta, T. Yang, P. Wang et al., "Local learning matters: Rethinking data heterogeneity in federated learning," in CVPR, 2022, pp. 8397-8406.
- [30] Q. Li, B. He, and D. Song, "Model-contrastive federated learning," in CVPR, 2021, pp. 10713-10722.
- [31] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in CVPR, 2015, pp. 815-823.
- [32] D. Yao, W. Pan, Y. Dai et al., "Local-global knowledge distillation in heterogeneous federated learning with non-iid data," arXiv preprint arXiv:2107.00051, 2021.
- G. Hinton, O. Vinyals, J. Dean et al., "Distilling the knowledge in a [33] neural network," arXiv preprint arXiv:1503.02531, vol. 2, no. 7, 2015.
- R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in CVPR, vol. 2, 2006, pp. 1735–1742.
- [35] S. P. Karimireddy, M. Jaggi, S. Kale et al., "Mime: Mimicking centralized stochastic algorithms in federated learning," arXiv preprint arXiv:2008.03606, 2020.
- [36] Y. LeCun, L. Bottou, Y. Bengio et al., "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278-2324, 1998
- [37] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." J Mach Learn Res, vol. 9, no. 11, 2008.
- [38] A. Paszke, S. Gross, F. Massa et al., "Pytorch: An imperative style, high-performance deep learning library," NIPS, vol. 32, pp. 8026-8037, 2019.
- [39] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konečný, H. B. McMahan, V. Smith, and A. Talwalkar, "Leaf: A benchmark for federated settings," arXiv preprint arXiv:1812.01097, 2018.

- [40] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," arXiv preprint arXiv:1708.07747, 2017.
- [41] G. Cohen, S. Afshar, J. Tapson *et al.*, "Emnist: Extending mnist to handwritten letters," in *IJCNN*, 2017, pp. 2921–2926.
- [42] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)," URL http://www. cs. toronto. edu/kriz/cifar. html, vol. 5, no. 4, p. 1, 2010.
- [43] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010.
- [44] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *NIPS*, vol. 25, 2012.
- [45] J. Wang, V. Tantia, N. Ballas *et al.*, "Slowmo: Improving communication-efficient distributed sgd with slow momentum," *arXiv* preprint arXiv:1910.00643, 2019.
- [46] I. Sutskever, J. Martens, G. Dahl *et al.*, "On the importance of initialization and momentum in deep learning," in *ICML*, 2013, pp. 1139–1147.