计算机科学
COMPUTER SCIENCE

基于粒子群优化的面向数据异构的联邦学习方法

徐奕成, 戴超凡, 马武彬, 吴亚辉, 周浩浩, 鲁晨阳

引用本文

徐奕成, 戴超凡, 马武彬, 吴亚辉, 周浩浩, 鲁晨阳. 基于粒子群优化的面向数据异构的联邦学习方法[J]. 计算机科学, 2024, 51(6): 391-398.

XU Yicheng, DAI Chaofan, MA Wubin, WU Yahui, ZHOU Haohao, LU Chenyang. Particle Swarm Optimization-based Federated Learning Method for Heterogeneous Data [J]. Computer Science, 2024, 51(6): 391-398.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.230400182

基于粒子群优化的面向数据异构的联邦学习方法

徐奕成　戴超凡　马武彬　吴亚辉　周浩浩　鲁晨阳
国防科技大学信息系统工程重点实验室　长沙 410073
(xuyicheng18@nudt.edu.cn)

摘　要　联邦学习是一种新兴的面向隐私保护的分布式机器学习框架,其核心特点是能够在不获取客户端原始数据的条件下实现分布式机器学习。客户端利用本地数据进行模型训练,然后将模型参数上传至服务端进行聚合,从而确保客户端数据始终得到保护。在此过程中,存在频繁的参数传输导致的通信成本高昂问题和各客户端所拥有的非独立同分布异构数据问题,两者严重制约了联邦学习的应用。针对上述问题,提出了一种基于粒子群优化的面向数据异构的联邦学习方法——FedPSG,将客户端传输到服务器的数据形式由模型参数转变为模型分值,在每轮训练中只需要少部分客户端向服务器上传模型参数,从而降低通信成本;同时,提出了一种模型再训练策略,使用服务器数据对全局模型进行二次迭代训练,通过缓解数据异构问题对联邦学习的影响来进一步提升模型性能。模拟不同的数据异构环境,在 MNIST,FashionMNIST 与 CIFAR-10 数据集上进行实验,结果表明 FedPSG 能够有效提高模型在不同数据异构环境下的准确率,并且验证了模型再训练策略能有效解决客户端数据异构问题。

关键词:联邦学习;粒子群算法;通信成本;数据异构;隐私保护
中图分类号　TP301

Particle Swarm Optimization-based Federated Learning Method for Heterogeneous Data

XU Yicheng,DAI Chaofan,MA Wubin,WU Yahui,ZHOU Haohao and LU Chenyang
National Key Laboratory of Information Systems Engineering,National University of Defense Technology,Changsha 410073,China

Abstract　Federated learning is an emerging privacy-preserving distributed machine learning framework,whose core feature is the ability to implement distributed machine learning without access to the client's raw data. The client uses local data for model training and then uploads the model parameters to the server for aggregation,thus ensuring that the client data is always protected. In this process,there are problems of high communication costs due to frequent parameter transfers and non-independent homogeneous heterogeneous data owned by each client,both of which severely limit the application of federated learning. To address these problems,FedPSG,a federated learning method based on particle swarm optimization for data heterogeneity,is proposed to reduce the communication cost by changing the form of data transferred from the client to the server from model parameters to model scores,so that only a small number of clients need to upload model parameters to the server in each training round. Meanwhile,a model retraining strategy is proposed to use the server data to train the global model for a second iteration,further improving the model performance by mitigating the impact of data heterogeneity issues on federated learning. Simulating different data heterogeneous environments,experiments are conducted on MNIST,FashionMNIST and CIFAR-10 datasets. The results show that FedPSG can effectively improve the accuracy of the model in different data heterogeneous environments,and verify that the model retraining strategy can effectively solve the client-side data heterogeneity problem.

Keywords　Federated learning,Particle swarm algorithm,Communication cost,Data heterogeneity,Privacy protection

1　引言

随着大数据技术的快速发展,机器学习在科学研究和商业决策中的应用越来越广泛。具有高表现力的数据能被用于构建更为复杂、准确的机器学习模型,进而为人们带来更加精准的服务和更好的决策支持。然而,数据应用仍存在问题。一方面,随着智能设备的普及和互联网的兴起,每天产生的数据量呈爆炸式增长,其中包括照片、语音等各类数据。另一方面,传统的机器学习依靠主服务器来存储数据和训练模型,但大部分数据都散布在各种移动终端上,导致数据聚合和模型训练面临着巨大的挑战。此外,世界各国出台关于数据隐私保护的法律法规,例如新加坡于 2019 年通过的《个人数据

到稿日期:2023-04-22　返修日期:2023-09-11
基金项目:国家自然科学基金面上项目(61871388)
This work was supported by the General Program of National Natural Science Foundation of China(61871388).
通信作者:马武彬(wb_ma@nudt.edu.cn)

保护法》与中国于2021年发布的《数据安全法》。这些法规的建立将有助于保护数据安全，但也制约着机器学习的发展，导致海量数据难以被挖掘与利用。

在这种背景下，人们开始将注意力从数据聚合转向模型聚合。联邦学习[1]是一种新的分布式机器学习框架，该框架可以在数据留在客户端本地而不上传至服务器的情况下，训练出一个良好的全局模型[2]。在联邦学习的训练过程中，每个客户端从服务器下载全局模型后，利用本地数据对模型进行训练，确保数据不离开本地，然后将训练后的本地模型上传至服务器，由服务器执行模型的聚合更新。通过迭代训练，得到一个性能良好的全局模型。联邦学习类似于传统的分布式机器学习，学习任务分配给多个客户端来完成，但区别在于，联邦学习的训练数据不是由服务器发送给各个客户端，每个客户端的训练数据是私有的本地数据。因为数据始终保存在本地，不会上传给服务器，也不会与其他客户端共享，所以数据的隐私与安全可以得到保障。

然而，联邦学习仍面临着一些挑战。首先，联邦学习在学习训练的过程中需要服务器与客户端之间的频繁参数传输。当下使用最多的是神经网络模型，模型的参数数量较大，这将导致联邦学习的模型训练需要高昂的通信成本[3]。其次，现实中各个客户端拥有的数据通常是非独立同分布的(Non-IID)[4]，即存在数据异构问题，这会导致联邦学习训练得到的模型精度下降，甚至训练无法达到收敛[5]。

针对以上问题，本文通过对联邦学习与粒子群优化算法的研究，提出了FedPSG，实现即使在客户端数据异构的情况下，联邦学习仍能保持低通信成本并且训练出相对高质量的全局模型。与以往提出的联邦学习算法不同，FedPSG将客户端模型分值作为通信数据，并采用Top-K最优策略[6]，在一轮通信中只需要K个客户端向服务器发送模型参数。此外，该方法在服务器端设置了模型再训练策略，构建能体现总体训练数据分布情况的服务器数据D_m，再使用D_m对模型进行二次迭代训练。本文在本地模拟了客户端数据异构的情况，实验证明FedPSG可以有效提高模型在不同数据异构环境下的准确率。同时，通过实验验证了模型再训练策略能有效解决数据异构问题。综上，本文的主要贡献如下：

1)结合粒子群优化算法与联邦学习训练过程，提出了FedPSG，将联邦学习的通信数据形式由模型参数变为模型分值，并通过Top-K最优策略选择客户端模型。对比传统联邦学习算法，FedPSG可以有效降低通信成本。

2)提出模型再训练策略。构建服务器数据，并在此基础上进行模型的二次训练优化，不仅缓解了数据异构问题对联邦学习的影响，还提升了模型的准确率。此外，该策略还能实现以可控的方式指导联邦模型的优化。

3)在MNIST，FashionMNIST与CIFAR-10数据集上模拟不同的数据异构情况，对比分析了FedPSG，FedAvg和FedShare算法的准确率。实验结果证明，由FedPSG得到的模型具有更高的准确率，并且模型再训练策略能有效解决数据异构问题；在数据异构情况下，设计服务器数据量对FedPSG模型准确率的影响实验，结果证明，服务器数据量越大，FedPSG模型的准确率越高，模型准确率在数据异构条件改变时的鲁棒性越强。

2 相关工作

McMahan等在提出联邦学习的同时，考虑到模型上传和下载速度不匹配，指出服务器和客户端之间的通信次数应该尽可能少，因此提出了FedAvg算法[1]。该算法通过在每一轮通信时增加每个客户端的计算量来减少上传次数，提高通信效率。受到FedAvg算法的启发，Nishio等[7]构建了FedCS框架，该框架在每一轮模型训练中通过选择客户端，使得服务器可以在有限的时间范围内聚合尽可能多的客户端进行模型更新，有效地加快了训练速度。Yurochkin等[8]设计了一种基于神经网络的贝叶斯非参数联邦学习框架，可以在不需要额外参数的情况下将本地模型聚合到联邦模型中，从而避免不必要的通信轮次。实验表明，该框架最少只需要一轮通信就可以获得满意的全局模型。然而，以上算法或框架在处理较大规模的Non-IID数据时收敛速度较慢，在极端的数据异构情况下甚至无法收敛。

Zhao等[9]提出了一种通过牺牲部分隐私来提升模型性能的策略(FedShare)，通过构建一个全局共享的数据集G来减小数据异构问题给联邦学习带来的影响。该策略要求数据集G能体现整体数据的分布情况，将数据集G存储在中央服务器中，并按照一定比例将G的一部分发送到每个客户端。在训练过程中，中央服务器使用G对全局模型进行预训练，客户端结合本地数据与G的子集数据对本地模型进行训练。在CIFAR-10数据集上的实验结果表明，全局共享5%的数据就可以将模型测试精度提高约30%。文献[10]与文献[11]也使用了数据共享的方式，与FedShare不同的是，它们将一部分数据在训练前从客户端发送到服务器。但是，通过数据共享缓解Non-IID问题有着明显的缺点：将部分数据集变为共享数据集的方式与联邦学习保护数据安全的初衷相矛盾。

近年来，也有许多研究通过改进选择客户端的策略来减小数据异构问题给联邦学习带来的影响。Wang等[12]基于强化学习提出了FAVOR，FAVOR框架通过学习主动选择每轮通信中的最佳客户端子集来加速和稳定联邦学习过程，以抵消Non-IID数据导致的训练偏差。Cai等[13]提出了一种动态样本选择优化算法，该算法在梯度迭代过程中根据本地可用数据大小动态地选择训练样本大小。对于Non-IID数据分布下的客户端，采用先分类后筛选的方法，选择最适合参加训练的客户端。Xu等[14]利用模型分割技术对数据分布进行分类，先确定数据分布与模型训练质量之间的关系，再选择有利于全局模型收敛的客户端参与训练。以上数据选择的方案不会改变数据和客户端本身，同时可以选择最有利于训练的客户端参与每一轮的联邦学习训练。然而，数据选择的方案会导致一部分算力较弱或者网络不稳定的客户端不能参与全局模型的训练，进而加剧联邦学习的不公平性。此外，选择客户端的方式会造成部分数据无法参与训练，导致模型精度下降。

对于复杂的优化问题或者非凸问题，粒子群算法在许多应用中表现出了具有较快的收敛速度与较好的收敛结果等优点[15]。近年来，也涌现了较多结合机器学习与粒子群算法的研究，主要是将粒子群算法用于优化神经网络的超参数[16-18]，

目的是提高模型在分类任务中的准确率。在分布式学习环境下，只有少数文献涉及了粒子群算法与联邦学习的结合应用。Qolomany 等[19]提出了一种基于粒子群优化的技术来优化联邦学习环境中客户端机器学习模型的超参数，但是粒子群优化技术并不用于联邦学习的训练过程。Park 等[20]提出了 FedPSO 算法，将粒子群算法与联邦学习的训练过程相结合，但 FedPSO 算法只是将粒子群算法与联邦学习简单结合，把客户端数据分布假定为理想化的独立同分布，并且该算法在 Non-IID 数据条件下的表现不稳定。

3 基于粒子群优化的面向数据异构的联邦学习方法

3.1 基本理论

3.1.1 联邦学习机制

联邦学习系统通常由一台服务器和多个客户端组成。FedAvg 算法是最典型的联邦学习算法，其思路是将运行随机梯度下降的各个客户端与运行模型平均计算的服务器结合起来[1]。如图 1 所示，在每一轮训练中，客户端下载全局模型并利用其本地数据训练模型，然后将本地模型参数上传至服务器。服务器负责协调各客户端共同训练，并通过聚合本地模型参数更新全局模型。

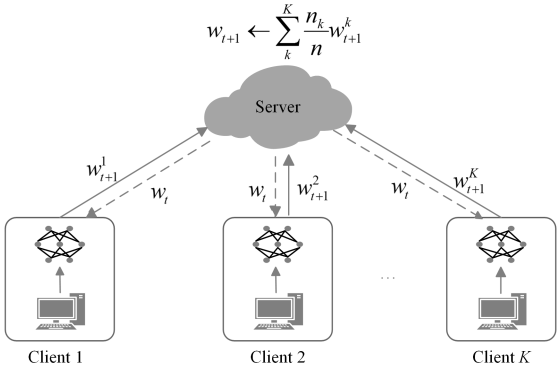


图 1 联邦学习的训练过程

Fig. 1 Training process of federated learning

3.1.2 粒子群优化算法

Kennedy 等[21]于 1995 年提出了粒子群算法。该算法受到自然界鸟类和鱼群的启发，可以同时优化多个变量，具有易于实现、可扩展性强、鲁棒性强、快速收敛和简单的数学运算等优点。这些优点使得粒子群算法在计算机上解决复杂问题时所需的内存更小，但解决问题的速度更快。

粒子群算法由一组粒子组成，每个粒子代表问题的一个解决方案。每个粒子都有参数：位置 x 和速度 v。为了找到全局最佳值，粒子之间相互通信并共享自己的局部最佳解（p_i）。每个粒子将全局最佳解（p_g）设置为各个粒子 p_i 的最大值，即 $p_g=\max(p_1,p_2,\cdots,p_N)$，$N$ 是粒子群的规模。在一轮迭代中，粒子 i 速度的更新会同时受到 p_i 和 p_g 的影响，更新过程如下：

$$v_i=\alpha v_i+c_1 r_1(p_i-x_i)+c_2 r_2(p_g-x_i) \tag{1}$$

其中，α 是代表惯性权重的常数，p_i 是粒子 i 的最佳解，p_g 是全局最佳解，c_1 是 p_i 的加速度常数，c_2 是 p_g 的加速度常数，r_1 和 r_2 的值是 0 到 1 之间的随机数。获取速度 v_i 后，粒子 i 对自身位置进行更新：

$$x_i=v_i+x_i \tag{2}$$

3.2 FedPSG

3.2.1 联邦粒子通信

在传统的联邦学习框架中，模型聚合的前提条件是服务器收到一定比例的客户端上传的模型参数。然而，该过程极易受到模型参数量与客户端网络情况的影响。例如，如果联邦学习训练的是一个 VGG16[22]模型，那么客户端在每一轮训练中需要传输的参数量将高达 1.3 亿。即使是轻量型的 MobileNet 系列，MobileNetV1 模型[23]仍有 400 多万的参数量。在需要传输如此大的参数量的背景下，一旦联邦学习通信过程中客户端的网络情况不稳定，整个联邦学习的训练速度都会受到影响。由此看来，减少联邦学习通信过程中传输的数据量很有必要。

为降低通信成本，提高联邦学习训练过程的鲁棒性，本文提出了 FedPSG，该算法改变了客户端与服务器之间传递的数据形式，具体实现方式是将传递数据的形式转变为客户端模型的分值（本文设定为准确率），而不是数据量较大的模型参数。在训练开始前，服务器向各个客户端发送评分数据集 D_s，D_s 是服务器数据 D_m 的独立同分布子集，D_m 的选择将在 3.2.2 节阐述。如图 2 所示，与 FedAvg 算法传递模型参数方式不同的是，在 FedPSG 的通信过程中，客户端向服务器上传局部模型的历史最佳分值。服务器在取得各个客户端传递的分值后进行对比，通过采取 Top-K 最优策略，确定在评分数据集上表现最好的前 K 个客户端，然后请求获取这些客户端的模型参数，再通过参数平均的方式更新这一轮的全局最佳模型。详细过程如算法 1 所示。因此，在一轮通信中，FedPSG 框架下的服务器只需要 K 个客户端的模型参数，不要求所有客户端发送模型参数，这样可以大大减少通信成本，提高通信效率，进而提高联邦学习训练过程的稳定性。

算法 1 FedPSG

输入：客户端数量 N；Top-K 最优策略参数 K；通信次数 T；本地迭代次数 E；学习率 η；惯性权重 α；加速因子 c_1 和 c_2

输出：全局模型 w_T

1. initializew_0，gbest，pbest$_i$，L_gid /* 1—13 是服务器执行的操作 */
2. for each round t from 1 to T do
3. for each client n from 1 to N in parallel do
4. pbest$_k$←ClientUpdate(n，w_t)
5. L_pbest←[pbest$_1$，…，pbest$_N$]
6. sort(L_pbest) /* 执行降序排序 */
7. pbest←L_pbest[1，K]
8. L_gid←gid of pbest
9. L_w_t←model of the client L_gid
10. w_t←the mean of all numbers in L_w_t
11. if gbest<score(w_t) then
12. gbest←score(w_t)
13. $w_t \leftarrow w_t-\eta\nabla l(w_t,D_m)$ /* D_m为服务器拥有的数据集 */
14. initialize w，V，w^{pbest} /* 14—23 是客户端执行的操作 */
15. ClientUpdate(n，w_t)：
16. β←(split D_n into batches of size B) /* D_n为客户端 n 拥有的数据集 */

17. for each weight layer $V_l \in V$ do
18. $V_l \leftarrow \alpha \cdot V_l + c_1 r_1 (w^{pbest} - V_l) + c_1 r_1 (w_t - V_l)$
19. $w \leftarrow w + V$
20. for each client epoch i from 1 to E do
21. for batch $b \in \beta$ do
22. $w \leftarrow w - \eta \nabla l(w, b)$
23. return $pbest_n$ to server / * $pbest_n$ 为客户端 n 的模型在评分数据集上准确率的最大值 * /

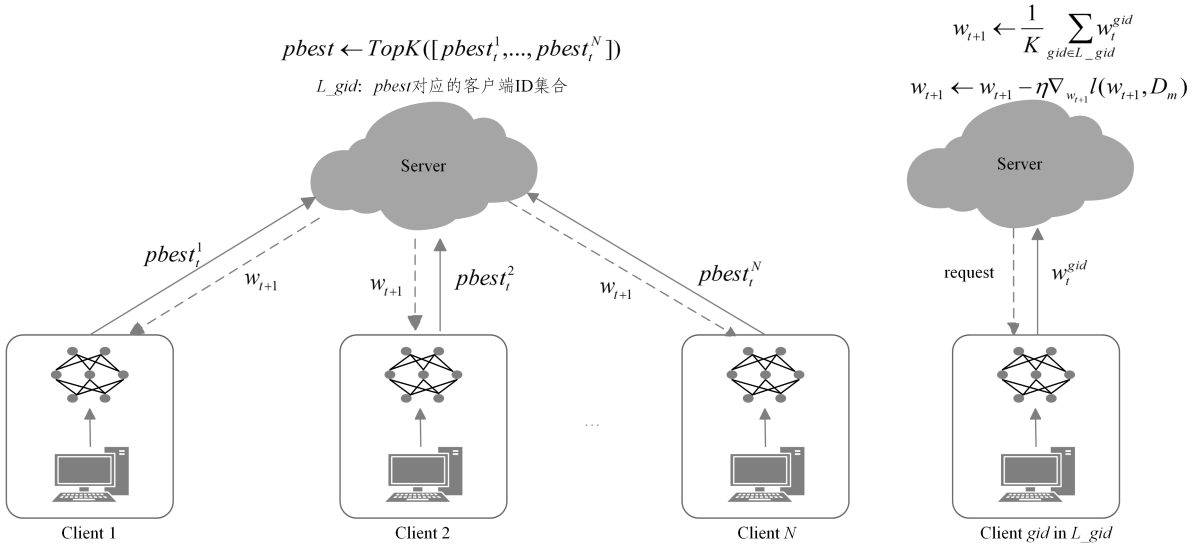


图 2 FedPSG 的训练过程

Fig. 2 Training process of FedPSG

为发挥全局最佳模型与局部最佳模型对客户端模型训练的指导作用，FedPSG 中客户端神经网络模型的更新包括一轮粒子更新与 E 轮本地迭代。粒子更新过程如式(3)和式(4)所示：

$$V_l^t = \alpha \cdot V_l^{t-1} + c_1 r_1 (w_p - V_l^{t-1}) + c_2 r_2 (w_g - V_l^{t-1}) \quad (3)$$

$$w_t = V_t + w_{t-1} \quad (4)$$

其中，w_p 表示该客户端的局部最佳模型；w_g 表示全局最佳模型；w_t 表示第 t 轮通信时的客户端模型；V_l^t 表示神经网络模型第 l 层的参数，所有层的参数构成了 V_t；α 表示惯性权重；r_1 和 r_2 是[0,1]之间的随机值；c_1 和 c_2 分别表示 w_p 与 w_g 的加速因子，c_2 越大，模型训练将更偏向全局模型，c_1 越大，模型训练将更偏向局部模型。

3.2.2 模型再训练策略

在 FedPSG 框架中，Top-K 最优策略影响着联邦学习的训练结果。K 的取值越小，表示在每一轮训练中服务器接收到的客户端模型越少，需要的通信成本也越少。然而，选择部分客户端模型的方式会导致另一部分客户端未参与全局模型的训练，部分在评分数据集上表现较好的客户端会主导全局模型的训练，这将造成联邦学习的不公平性。当 K 很小，如 $K=1$ 时，服务器只接受一个客户端的模型，然而单个客户端的数据不能代表所有客户端的数据，尤其是在 Non-IID 数据环境下。如果简单地把一个客户端模型设定为全局模型，将导致联邦学习训练全局模型使用的数据分布与真实数据分布不一致[4]，进而造成模型精度的下降。考虑以下联邦学习场景：使用 MNIST 数据集作为训练数据，但处于极端的数据非独立同分布条件下，即每个客户端只拥有一类数据。在这种情况下，单个客户端训练得到的本地模型的泛化效果显然很差。如果将单个客户端模型作为全局模型发送给各个客户端，由于全局模型实际上只在一类数据上进行训练，因此该全局模型只能改进客户端模型对某一类标签数据的识别能力，经过上述训练过程最终得到的全局模型也将缺乏泛化能力。K 的取值越大，服务器在一轮训练中接收到的客户端模型越多，这有助于充分利用客户端数据，训练更具泛化能力的全局模型。然而，当 K 的取值很大，如 K 等于客户端数量时，FedPSG 需要的通信成本将和 FedAvg 一样多。

为了降低通信成本、减少服务器需要接收的客户端模型数量，同时缓解 Non-IID 问题给联邦学习训练带来的影响，FedPSG 在服务器端设计了模型再训练策略：在每一轮通信过程中，服务器收到客户端模型后，利用服务器数据 D_m 对模型进行再训练：

$$w_t = w_t - \eta \nabla l(w_t, D_m) \quad (5)$$

其中，w_t 表示在第 t 轮通信过程中服务器收到的客户端模型，η 是学习率，D_m 应该要体现所有客户端总体训练数据的数据分布情况。本文中，D_m 是总体训练数据 D 的一个独立同分布子集，D_m 数据量与 D 数据量的比值为 γ。在实践中，可以通过一些用户自愿共享的数据或者招募一些志愿者参与内部测试来获得 D_m。通常情况下，服务器拥有较好的信息资源，可以根据历史数据和自身情况构建 D_m。这些构建 D_m 的方法不违反联邦学习的隐私保护原则，并已在之前的一些研究中得到了应用[24-25]。

从另一个角度来看，模型再训练策略提供了一种解决联邦学习不公平性问题的方法。服务器数据 D_m 并不需要与总体训练数据 D 有必然的联系，相反，应该根据真实目的选择 D_m。例如，D 可能包含一些种族、性别或者是财富上的歧视现象，但我们可以针对性地构建一个 D_m 来指导联邦学习模型的优化，使其朝着无偏见和公平的方向发展。

如 3.2.1 节所述，在 FedPSG 训练开始前，服务器需要向客户端发送评分数据集 D_s。因为在 D_s 上准确率最高的客户

端才能将模型发送给服务器，所以 D_s 实际上代表着 FedPSG 客户端模型优化的方向。由于服务器可以根据目标构建 D_m，为了使服务器与客户端训练目标一致，本文将 D_s 设置为 D_m 的一个独立同分布子集，D_s 数据量与 D_m 数据量的比值为 δ。

4 实验分析

4.1 实验环境

实验所用环境是配置了 Intel(R) Xeon(R) Platinum 8350C CPU@2.60 GHz 处理器以及 NVIDIA GeForce RTX 3090 显卡的 Ubuntu 系统。实验通过 Python 语言实现，并且使用了 PyTorch 神经网络框架。采用的 Python 版本为 3.8，使用的 PyTorch 版本为 1.11，实验代码已公布在 GitHub[1)]。

4.2 实验数据

本文实验采用 MNIST 数据集[26]、FashionMNIST 数据集[27]以及 CIFAR-10 数据集[28]对 FedPSG 进行评估，并在不同的数据异构情况下将 FedPSG 与 FedAvg，FedShare 算法进行对比。同时还设置了服务器数据对 FedPSG 的影响实验，对比了在不同 γ 值情况下 FedPSG 在 MNIST 数据集、FashionMNIST 数据集以及 CIFAR-10 数据集上的准确率。

MNIST 数据集由 60000 个训练样本和 10000 个测试样本组成，其中每一个样本都是一张 28×28 像素的灰度手写数字图片。FashionMNIST 数据集涵盖了来自 10 种类别的共 70000 个不同商品的正面图片，划分为 60000 个训练样本与 10000 个测试样本，其中每一个样本都是一张 28×28 像素的灰度图片。CIFAR-10 数据集由 50000 个训练样本和 10000 个测试样本构成，每一个样本都是一张 32×32 像素的彩色图片。相比 MNIST 数据集与 FashionMNIST 数据集中的样本，CIFAR-10 中的样本是现实世界中真实的物体，不仅噪声很大，而且物体的特征、大小都不尽相同，因此识别难度更大。为模拟联邦学习的数据分布环境，本文设置了 100 个客户端作为训练节点，并根据文献[4]中的数据集划分方式，将训练数据划分为 IID，Non-IID(1)和 Non-IID(2)。

1)IID：将训练数据随机均匀地分配给每个客户端。以 MNIST 数据集的划分为例，数据集中每一类数据有 6000 个训练样本，将数据集的 10 个类别的数据分别随机均匀地分发给 100 个客户端，最终每个客户端分配到的数据量都是 600，且都能拥有 10 类标签的数据。这样的划分方式可以保证每个客户端的训练数据集具有一定的多样性和代表性，有利于提高模型的泛化能力。

2)Non-IID(1)：每个客户端只拥有 1 类标签的数据。以 MNIST 数据集的划分为例，首先将数据集按照标签分成 10 个组，每组数据平均地拆分为 100 个数据切片。然后，每个客户端从随机一组中取 10 个数据切片，构成客户端训练数据。

3)Non-IID(2)：每个客户端只拥有两类标签的数据。以 MNIST 数据集的划分为例，首先将数据集按照标签分成 10 个组，每组数据平均地拆分为 200 个数据切片。然后，每个客户端从随机的两组中各取 10 个数据切片，构成客户端训练数据。

实验模型采用卷积神经网络模型，并参考文献[1]中的网络结构设置：采用两个 5×5 卷积层，每个卷积层后面接着一个 2×2Max 池化层，之后接着 3 个全连接层，最终输出一个 10 维向量。

4.3 参数设置

对于 MNIST 数据集与 FashionMNIST 数据集，batch_size 设置为 20，本地迭代次数 E 设置为 10，学习率 $\eta=0.001$。对于 CIFAR-10 数据集，batch_size 设置为 20，本地迭代次数 E 设置为 10，学习率 $\eta=0.01$。FedAvg 算法选择客户端的比例 $C=1.0$。FedShare 是 FedAvg 的改进算法，主要研究的是联邦学习中的 Non-IID 问题，该算法的参数设置参考了文献[9]，从每一类数据中取 1000 个样本组成全局共享数据集 G，再分配给每个客户端比例为 β 的全局共享数据集，且 $\beta=0.05$。FedPSG 中 D_s 数据量与 D_m 数据量的比值 δ 为 0.2，Top-K 最优策略参数 K 取值为 10。

4.4 实验性能分析

4.4.1 客户端数据异构下的联邦学习实验性能分析

为测试 FedPSG 在客户端数据异构条件下的性能，实验设置了 IID，Non-IID(1)与 Non-IID(2)环境，并在 MNIST，FashionMNIST 与 CIFAR-10 数据集上对比 FedPSG，FedAvg，FedShare 与 FedPS 算法的准确率。其中，FedPS 算法是 FedPSG 去掉模型再训练策略后的简化版，设定服务器数据 D_m 的数据量与客户端训练数据 D 的数据量比值 $\gamma=0.2$。IID 的数据是独立同分布的，不存在客户端数据异构情况。Non-IID(1)与 Non-IID(2)的数据是非独立同分布的，但是 Non-IID(1)数据的异构程度更高。实验结果如图 3 和表 1 所示。

可以观察到，随着客户端数据异构程度的提高，FedAvg 算法在 MNIST，FashionMNIST 与 CIFAR-10 数据集上的准确率下降显著。经过 100 轮训练后，当数据异构情况从 IID 变为 Non-IID(2)时，FedAvg 算法在 MNIST，CIFAR-10 与 FashionMNIST 数据集的准确率分别下降了 15%，20.08%和 28.65%；当数据异构情况从 IID 变为 Non-IID(1)时，FedAvg 算法在 MNIST，CIFAR-10 与 FashionMNIST 数据集上的准确率分别下降了 57.53%，35.89%和 32.76%。由此可见，FedAvg 模型的准确率随着数据异构程度的提升而降低，这验证了客户端数据异构情况对传统联邦学习的训练影响极大。

相比 FedAvg，FedShare 与 FedPSG 在 Non-IID 数据条件下的表现更稳定。尤其是在 MNIST 数据集上，当数据异构条件由 IID 变为 Non-IID(1)或者 Non-IID(2)时，FedShare 与 FedPSG 准确率的变化都不超过 2%。但是 FedPSG 在 CIFAR-10 与 FashionMNIST 数据集上的准确率明显高于 FedShare，并且随着客户端数据异构程度的提高，FedPSG 准确率的下降程度低于 FedShare。如表 1 所列，在 CIFAR-10 与 FashionMNIST 数据集上，随着数据异构程度的提高，FedPSG 准确率的变化不超过 2%。而 FedShare 在数据异构条件从 IID 变为 Non-IID(1)时，在 CIFAR-10 与 FashionMNIST 数据集上的准确率分别下降了 11.8%和 5.18%。

1) https://github.com/xyccjd/FedPSG

可见，本文提出的 FedPSG 在应对数据异构问题时是有效的，并且效果要优于一般的改进算法。

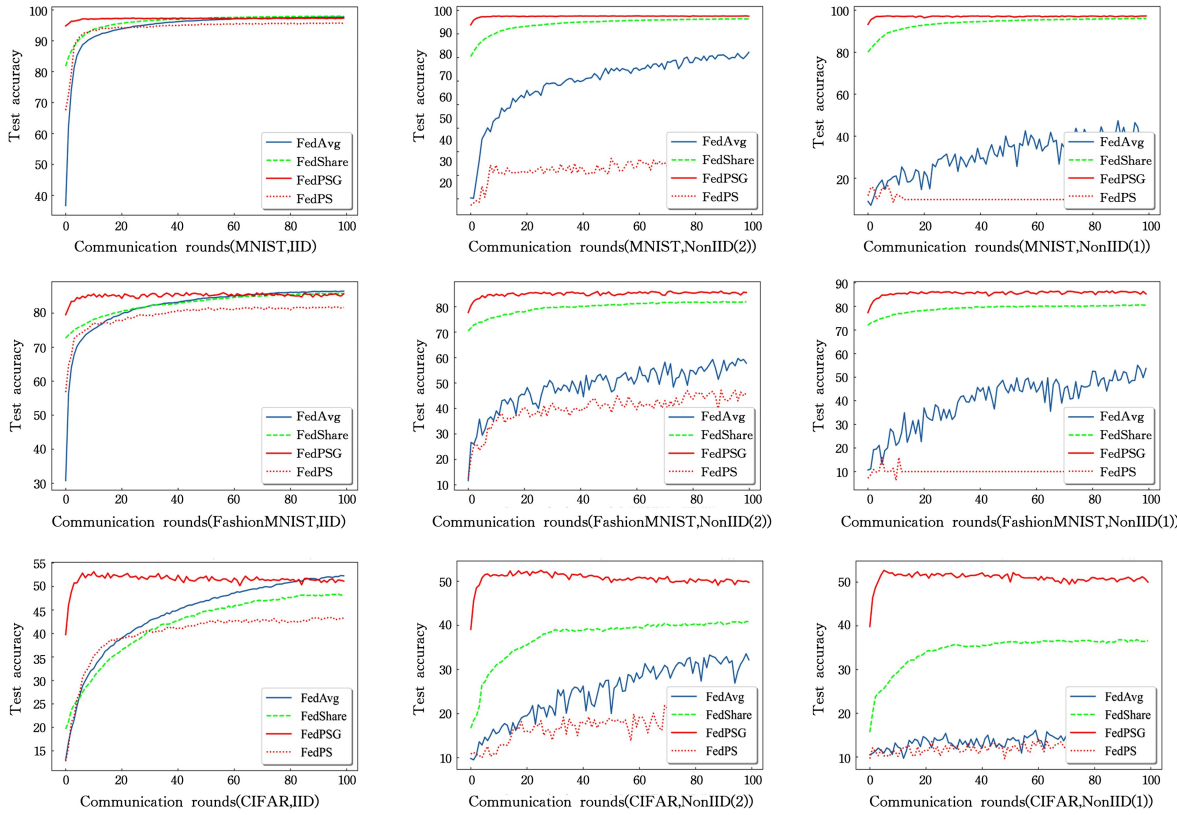


图 3 3 种客户端数据异构条件下 FedPSG，FedShare，FedAvg 与 FedPS 在 CIFAR-10，MNIST，FashionMNIST 数据集上的准确率

Fig. 3 Accuracy of FedPSG, FedShare, FedAvg and FedPS in CIFAR-10, MNIST and FashionMNIST under three heterogeneous conditions of client data

表 1 3 种客户端数据异构条件下 FedPSG，FedShare，FedAvg 与 FedPS 在 MNIST，CIFAR-10 与 FashionMNIST 数据集上经过 100 轮训练后的准确率

Table 1 Accuracy of FedPSG, FedShare, FedAvg and FedPS in MNIST, CIFAR-10 and FashionMNIST after 100 rounds of training under three heterogeneous conditions of client data

(%)

	MNIST			CIFAR-10			FashionMNIST		
	IID	Non-IID(1)	Non-IID(2)	IID	Non-IID(1)	Non-IID(2)	IID	Non-IID(1)	Non-IID(2)
FedPSG	97.33	**97.25**	**97.54**	51.16	**50.01**	**49.76**	85.71	**85.26**	**85.55**
FedShare	**97.94**	96.02	96.38	48.39	36.59	40.82	85.70	80.52	81.89
FedAvg	97.73	40.20	82.23	**52.28**	16.39	32.17	**86.46**	53.70	57.81
FedPS	95.52	9.80	39.64	43.28	13.30	16.85	81.70	10.00	46.11

如图 3 所示，在不同的数据异构条件下，FedPS 在 MNIST，CIFAR-10 与 FashionMNIST 数据集上的准确率都低于 FedPSG，甚至低于 FedAvg。随着客户端数据异构程度的提高，FedPS 在 MNIST，CIFAR-10 与 FashionMNIST 数据集上的准确率下降显著。当数据异构情况从 IID 变为 Non-IID(1)时，结果如表 1 所列，FedPS 在 MNIST 数据集上的准确率从 95.52%降至 9.8%，在 CIFAR-10 数据集上的准确率从 43.28%降至 13.3%，在 FashionMNIST 数据集上的准确率由 81.7%降至 10%。由于 FedPS 较之 FedPSG 只省去了模型再训练策略，且 FedPSG 在不同数据异构情况下均能保持较高的准确率，因此可以验证模型再训练策略能有效提高模型的准确率，并且提高联邦学习在数据异构情况下的训练质量。

4.4.2 D_m 数据量对 FedPSG 的影响实验性能分析

由 4.3 节可知，模型再训练策略有助于提升 FedPSG 在不同数据异构情况下的准确率。由于在执行该策略时所用数据是服务器数据 D_m，为进一步探究 D_m 对 FedPSG 的影响，实验设置了 5 种 γ 值，即 0.2，0.15，0.1，0.06，0.02，γ 值越接近 1，则 D_m 数据量越接近总体训练数据 D。在不同数据异构情况下对比了 FedPSG 在 MNIST，CIFAR-10 与 FashionMNIST 数据集上的准确率，结果如图 4 与表 2 所示。

可以观察到，在 Non-IID(1)或者 Non-IID(2)的数据异构条件下，随着 γ 值的增大，FedPSG 在 MNIST，CIFAR-10 与 FashionMNIST 数据集上的准确率不断提高，可知 D_m 数据量的增大有助于提高模型在数据异构情况下的精度。

经过 100 轮训练后，当数据异构条件由 IID 变为 Non-IID(1)时，模型($\gamma=0.20$)在 MNIST 数据集上的准确率从 97.33%降至 97.25%，降低了 0.08%，而模型($\gamma=0.02$)的准确率从 96.54%降至 92.03%，降低了 4.51%，可知在 MNIST 数据集上模型($\gamma=0.02$)准确率的下降程度大于模型($\gamma=$

0.20)；模型($\gamma=0.20$)在 CIFAR-10 数据集上的准确率从 51.16%降至 50.06%，降低了 1.1%，而模型($\gamma=0.02$)的准确率从 42.79%降至 34.57%，降低了 8.22%，可知在 CIFAR-10 数据集上模型($\gamma=0.02$)准确率的下降程度大于模型($\gamma=0.20$)；同样地，模型($\gamma=0.20$)在 FashionMNIST 数据集上的准确率从 85.71%降至 85.26%，降低了 0.45%，而模型($\gamma=0.02$)的准确率从 80.21%降至 76.94%，降低了 3.27%，即在 FashionMNIST 数据集上模型($\gamma=0.02$)准确率的下降程度大于模型($\gamma=0.20$)。可见，随着 γ 的减少，FedPSG 的模型精度在数据异构程度改变时的鲁棒性减弱。因此，为保持 FedPSG 在数据异构情况下训练结果的高质量，应尽可能增加务器数据 D_m 的数据量。

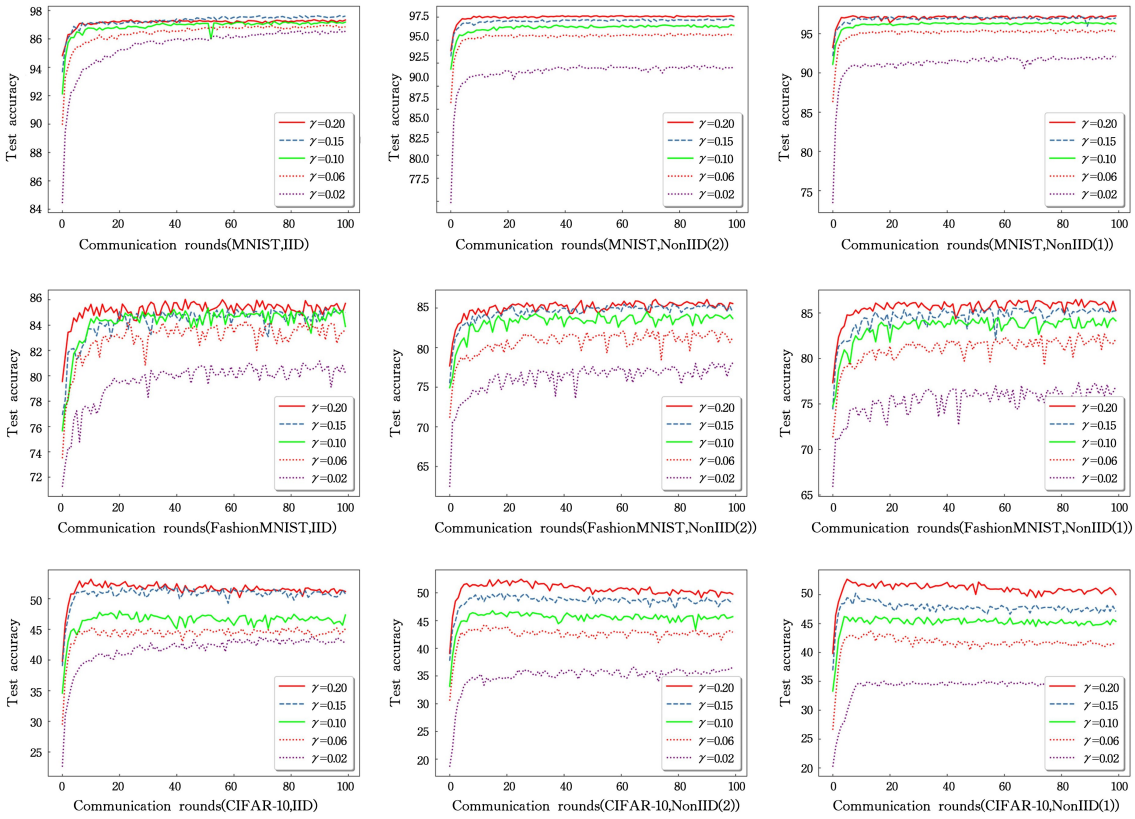


图 4　3 种客户端数据异构条件下 FedPSG 在 MNIST，CIFAR-10 和 FashionMNIST 数据集上的准确率随 γ 的变化

Fig. 4　Variation of FedPSG accuracy with γ on MNIST CIFAR-10 andFashionMNIST datasets under three heterogeneous conditions of client data

表 2　3 种客户端数据异构条件下 FedPSG 在 MNIST，CIFAR-10 和 FasionMNIST 数据集上经过 100 轮训练后的准确率随 γ 的变化

Table 2　Variation of FedPSG accuracy with γ after 100 rounds of training on MNIST，CIFAR-10 and FashionMNIST datasets under three heterogeneous conditions of client data

(%)

	MNIST			CIFAR-10			FashionMNIST		
	IID	Non-IID(1)	Non-IID(2)	IID	Non-IID(1)	Non-IID(2)	IID	Non-IID(1)	Non-IID(2)
$\gamma=0.20$	97.33	**97.25**	**97.54**	**51.16**	**50.01**	**49.76**	**85.71**	**85.26**	**85.55**
$\gamma=0.15$	**97.61**	97.01	97.19	50.78	47.10	48.17	84.09	85.22	84.44
$\gamma=0.10$	97.18	96.24	96.58	47.33	45.37	45.70	83.89	84.13	83.66
$\gamma=0.06$	96.85	95.34	95.57	45.07	41.51	42.88	83.64	82.13	81.26
$\gamma=0.02$	96.54	92.03	92.14	42.79	34.57	36.56	80.21	76.94	78.08

结束语　本文针对联邦学习中存在的通信成本高昂问题和客户端数据异构对模型精度产生消极影响的问题，提出了一种基于粒子群优化的面向数据异构的联邦学习方法——FedPSG。FedPSG 在联邦学习训练过程中，通过将通信数据形式由模型参数转变为模型分值，实现了在一轮通信中只需要少部分客户端向服务器发送模型参数，从而极大地降低了通信成本。此外，FedPSG 设置模型再训练策略，构建服务器数据 D_m，并在服务器端利用 D_m 对全局模型进行再训练，有效提高了全局模型的准确率，使得联邦学习在客户端数据异构情况下能保持模型的高质量。

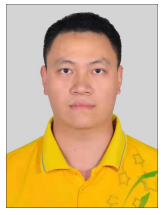
然而，本文的工作仍存在不足，为达到降低通信成本的目标，FedPSG 在使用 Top-K 最优策略时需要选取较少的客户端模型，因此 FedPSG 的训练结果非常依赖服务器数据的质量。如何在保证数据隐私的情况下构建高质量的服务器数据是未来的研究方向之一。

参 考 文 献

[1]　MCMAHAN B，MOORE E，RAMAGE D，et al. Communica-

tion-Efficient Learning of Deep Networks from Decentralized Data[C]// Artificial Intelligence and Statistics. PMLR, 2017: 1273-1282.
[2] RODRÍGUEZ-BARROSO N, JIMÉNEZ-LÓPEZ D, LUZÓN M V, et al. Survey on Federated Learning Threats: concepts, taxonomy on attacks and defences, experimental study and challenges[J]. Information Fusion, 2023, 90: 148-173.
[3] ALMANIFI O R A, CHOW C O, THAM M L, et al. Communication and computation efficiency in Federated Learning: A survey[J]. Internet of Things, 2023, 22: 100742.
[4] MA X, ZHU J, LIN Z, et al. A state-of-the-art survey on solving non-IID data in Federated Learning[J]. Future Generation Computer Systems, 2022, 135: 244-258.
[5] LI Z, SHARMA V, MOHANTY SP. Preserving Data Privacy via Federated Learning: Challenges and Solutions [J]. IEEE Consumer Electronics Magazine, 2020, 9(3): 8-16.
[6] CHEN M, BEUTEL A, COVINGTON P, et al. Top-K Off-Policy Correction for a REINFORCE Recommender System[C]// Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining(WSDM '19). 2019: 456-464.
[7] NISHIO T, YONETANI R. Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge[C]// 2019 IEEE International Conference on Communications (ICC 2019). 2019: 1-7.
[8] YUROCHKIN M, AGARWAL M, GHOSH S, et al. Bayesian Nonparametric Federated Learning of Neural Networks [J]. arXiv: 1905. 12022, 2019.
[9] ZHAO Y, LI M, LAI L, et al. Federated Learning with Non-IID Data[J]. arXiv: 1806. 00582, 2018.
[10] YOSHIDA N, NISHIO T, MORIKURA M, et al. Hybrid-FL for Wireless Networks: Cooperative Learning Mechanism Using Non-IID Data [C]// 2020 IEEE International Conference on Communications(ICC 2020). 2020: 1-7.
[11] TUOR T, WANG S, KO B J, et al. Overcoming Noisy and Irrelevant Data in Federated Learning[C]// 2020 25th International Conference on Pattern Recognition. 2021: 5020-5027.
[12] WANG H, KAPLAN Z, NIU D, et al. Optimizing Federated Learning on Non-IID Data with Reinforcement Learning[C]// IEEE Conference on Computer Communications (INFOCOM 2020). IEEE, 2020: 1698-1707.
[13] CAI L, LIN D, ZHANG J, et al. Dynamic Sample Selection for Federated Learning with Heterogeneous Data in Fog Computing [C]// 2020 IEEE International Conference on Communications (ICC 2020). 2020: 1-6.
[14] XU X, DUAN S, ZHANG J, et al. Optimizing Federated Learning on Device Heterogeneity with A Sampling Strategy[C]// 2021 IEEE/ACM 29th International Symposium on Quality of Service. 2021: 1-10.
[15] POLI R, KENNEDY J, BLACKWELL T. Particle swarm optimization: An overview [J]. Swarm Intelligence, 2007, 1(1): 33-57.
[16] SERIZAWA T, FUJITA H. Optimization of Convolutional Neural Network Using the Linearly Decreasing Weight Particle Swarm Optimization[J]. arXiv: 2001. 05670, 2020.
[17] WANG B, XUE B, ZHANG M. Particle Swarm Optimisation for Evolving Deep Neural Networks for Image Classification by Evolving and Stacking Transferable Blocks[C]// 2020 IEEE Congress on Evolutionary Computation. 2020: 1-8.
[18] SYULISTYO A R, PURNOMO D M J, RACHMADI M F, et al. Particle Swarm Optimization(PSO) For Training Optimization on Convolutional Neural Network(CNN)[J]. Jurnal Ilmu Komputer dan Informasi, 2016, 9(1): 52-58.
[19] QOLOMANY B, AHMAD K, AL-FUQAHA A, et al. Particle Swarm Optimized Federated Learning For Industrial IoT and Smart City Services[C]// 2020 IEEE Global Communications Conference(GLOBECOM 2020). 2020: 1-6.
[20] PARK S, SUH Y, LEE J. FedPSO: Federated Learning Using Particle Swarm Optimization to Reduce Communication Costs [J]. Sensors, 2021, 21(2): 600.
[21] KENNEDY J, EBERHART R. Particle swarm optimization [C]// Proceedings of ICNN'95 — International Conference on Neural Networks. 1995, 4: 1942-1948.
[22] SIMONYAN K, ZISSERMAN A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. arXiv: 1409. 1556, 2014.
[23] HOWARD A G, ZHU M, CHEN B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications [J]. arXiv: 1704. 04861, 2017.
[24] JEONG E, OH S, KIM H, et al. Communication-Efficient On-Device Machine Learning: Federated Distillation and Augmentation under Non-IID Private Data[J]. arXiv: 1811. 11479, 2018.
[25] YAO X, HUANG T, ZHANG R X, et al. Federated Learning with Unbiased Gradient Aggregation and Controllable Meta Updating[J]. arXiv: 1910. 08234, 2020.
[26] DENG L. The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web][J]. IEEE Signal Processing Magazine, 2012, 29(6): 141-142.
[27] HAN X, RASUL K, VOLLGRAF R. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms [J]. arXiv: 1708. 07747, 2017.
[28] KRIZHEVSKYA. Learning multiple layers of features from tiny images[J]. Handbook of Systemic Autoimmune Diseases, 2009, 1(4): 1-60.

XU Yicheng, born in 1998, postgraduate. His main research interests include federated learning and optimization algorithm.



MA Wubin, born in 1986, Ph. D, associate researcher. His main research interests include multi-objective optimization, micro-service and data mining.

(责任编辑:喻藜)