

黄勤凯, 龙伟, 胡灵犀, 等. 基于非独立同分布的联邦学习客户端选择方法[J]. 智能计算机与应用, 2026, 16(1): 178-184.  
DOI: 10.20169/j.issn.2095-2163.24040201

# 基于非独立同分布的联邦学习客户端选择方法

黄勤凯<sup>1</sup>, 龙伟<sup>1</sup>, 胡灵犀<sup>1</sup>, 蒋林华<sup>1</sup>, 张冠华<sup>2</sup>

(1 湖州师范学院 信息工程学院, 浙江 湖州 313000; 2 上海东滩智联网研究院, 上海 202162)

**摘要:** 联邦学习 (FL) 是一种分布式的机器学习方式, 无需将原始数据传输到服务器, 仅需要在本地设备上对模型进行训练。然而现实客户端数据具有异构性, 数据的异构性导致 FL 模型收敛缓慢。针对非独立同分布数据的问题, 首先研究提出了一种根据客户端贡献动态改变节点概率的节点选择方法 (FedDPCS), 该方法通过全局梯度和局部梯度筛选不利于全局更新的客户端, 然后在每一轮训练中根据客户端贡献改变每个节点的概率。其次, 从理论上分析了此方法相对于常用的 Federated Averaging (FedAvg) 算法的改进。最后通过实验证明, FedDPCS 在 CIFAR-10 数据集上比 FedAvg 算法提高 6.43%, 比 FedBn2 算法提高了 5.14%; 在 MNIST 数据集上比 FedAvg 算法提高了 5.24%, 比 FedBn2 算法提高了 4.36%。

**关键词:** 机器学习; 联邦学习; 非独立同分布; 节点选择

中图分类号: TP181

文献标志码: A

文章编号: 2095-2163(2026)01-0178-07

## Federated Learning client selection methods based on non-independent and identically distributed data

HUANG Qinkai<sup>1</sup>, LONG Wei<sup>1</sup>, HU Lingxi<sup>1</sup>, JIANG Linhua<sup>1</sup>, ZHANG Guanhua<sup>2</sup>

(1 School of Information Engineering, Huzhou University, Huzhou 313000, Zhejiang, China;

2 Shanghai Dongtan Intelligent Network Research Institute, Shanghai 202162, China)

**Abstract:** Federated Learning (FL) is a distributed machine learning approach that conducts model training directly on local devices without the need to transmit raw data to a central server. However, the inherent heterogeneity in real-world client data introduces challenges, leading to sluggish convergence of FL models. To address the issue of non-independent and identically distributed (non-i.i.d.) data, this paper firstly introduces a node selection method named Federated Dynamic Probability Client Selection (FedDPCS). This method dynamically adjusts selection probabilities based on client contributions, utilizing both global and local gradients to filter out clients less conducive to global updates. Subsequently, in each training round, the probabilities of each node are modified according to their respective contributions. The paper also provides the theoretical analysis, highlighting the improvements of FedDPCS compared to the commonly used Federated Averaging (FedAvg) algorithm. Finally, the experiment proves that FedDPCS improved 6.43% compared to FedAvg algorithm and 5.14% compared to FedBn2 algorithm on the CIFAR-10 dataset; on the MNIST dataset, FedDPCS improved 5.24% compared to FedAvg algorithm and 4.36% compared to FedBn2 algorithm.

**Key words:** machine learning; Federated Learning; non-iid data; node selection

## 0 引言

随着人工智能算力的高速发展, 机器学习作为海量数据的分析处理技术, 已经广泛运用于各个领域。然而机器学习技术目前面临着两大挑战: 首先是大量的用户数据难以得到保障, 隐私信息数据的泄露问题还有待解决<sup>[1]</sup>; 其次是数据孤岛问题<sup>[2]</sup>。

当数据完全隔离时, 势必会带来数据信息无法交流与整合的问题。为了解决上述的问题, 谷歌提出了联邦学习<sup>[3]</sup>。

目前基于联邦学习的研究成果持续涌现, 但是针对数据的增长, 研究发现, 每个客户端拥有的数据存在较大的差异, 非独立同分布的数据对于中心模型的训练会产生很大的负面影响。因此, 有研究者

**作者简介:** 黄勤凯 (1998—), 男, 硕士研究生, 主要研究方向: 智能信息处理技术; 龙伟 (1978—), 男, 博士, 讲师, 主要研究方向: 计算机应用, 图像处理, 人工智能等; 胡灵犀 (1991—), 女, 博士研究生, 主要研究方向: 光子准粒子和结构物质的相互作用, 光电子学太赫兹辐射源以及图像处理等; 蒋林华 (1977—), 男, 教授, 博士生导师, 主要研究方向: 计算机视觉, 人工智能, 自动驾驶等。

**通信作者:** 张冠华 (1975—), 男, 博士, 主要研究方向: 嵌入式技术, 物联网技术等研究。Email: 1138805223@qq.com。

收稿日期: 2024-04-02

提出选择有利于模型训练的参与者进行梯度聚合较为重要, 这类问题称为联邦学习客户端选择 (FRCS)<sup>[4-7]</sup>。

本文探讨了在非独立同分布数据下联邦学习的客户端选择问题, 提出了一种非独立同分布的联邦学习客户端选择机制。本文工作如下:

(1) 提出了一种非独立同分布联邦学习的客户端筛选机制, 主要根据客户端的局部梯度和中央服务器的全局梯度进行初步筛选。

(2) 基于上述筛选机制, 本文提出了一种通过计算客户端的贡献值动态地改变不同客户端被选中的概率, 提高对于全局模型有更高贡献的客户端被选中参与中央服务器进行聚合的概率。

(3) 通过实验, 在 MNIST 和 CIFAR-10 数据下, 与 FedAvg 以及 FedBn2 算法进行对比实验。实验结果表明, 与常用的 FedAvg 算法相比, 本文提出的方法在联邦学习模型的收敛速度方面有了很大的提升。

## 1 相关工作

近年来, 研究者对于联邦学习发展过程中产生的问题也提出了多种解决方法<sup>[8]</sup>。其中就包括联邦学习客户端选择问题。联邦学习的一些现有工作集中在服务器与客户端的通信成本方面, 研究发现通信成本是 FL 中的关键瓶颈<sup>[9]</sup>, 因此有研究者通过排除一部分客户端降低通信成本。Wang 等学者<sup>[10]</sup>提出了一种识别由不同数据分布引起的节点不相关更新的方法。Liu 等学者<sup>[11]</sup>提出了一种分层模式的联邦学习架构减少服务器和客户端之间的通信次数, 达到降低通信成本的目的。

同样为了提高联邦学习效率, 有研究者从算法角度考虑客户端的数据异构性问题。Wu 等学者<sup>[12]</sup>提出了一种基于每个客户端贡献的加权设计方法, 通过梯度信息反映局部更新与全局更新的相关性, 从而为参与节点分配权重。Nagalapatti 等学者<sup>[13]</sup>提出了一种选择具有相关数据的客户端框架。该框架为每个客户端的各数据块分配一个适应 GLM 状态的相关性分数, 通过相关性分数选择对全局模型更新更加有益的节点。

在联邦学习客户端的选择研究中, 全局模型的收敛速度是判定一个框架的重要指标, 因此有研究者着重于通过节点选择设计提高联邦学习的收敛速度。Cho 等学者<sup>[14]</sup>提出了一种基于局部损失的节点选择策略, 通过选择局部损失较大的客户端加快模型的收敛, 但是此方法存在较大的误差。Ren 等

学者<sup>[15]</sup>提出了一种根据本地更新的重要性和传输延迟分配客户端概率的方法。其中, 本地更新的重要性是通过本地梯度与全局梯度的差异来评估的。Rizk 等学者<sup>[16]</sup>设计了一种节点重要性采样, 每个节点被选择的概率和其本地梯度更新的范数成正比。

本文针对联邦学习中客户端的数据异构性, 设计了一种联邦学习的客户端选择机制, 该机制继承了上述方法从客户端的资源出发筛选出对于全局模型贡献更高的客户端。由于非独立同分布的数据对模型的质量呈正相关<sup>[17]</sup>, 当面对的本地数据差异性较大的客户端时, 采取通过本地训练获得的局部梯度与全局梯度的梯度方向判断该客户端是否有利于全局模型的收敛, 从而能够避免不利的局部更新对全局模型的影响, 确保模型的收敛速度; 整个过程中并不会收集客户端数据的任何信息, 也符合联邦学习保护用户隐私以及数据安全的要求。

## 2 联邦学习

### 2.1 基本原理

联邦学习在本质上是一种分布式系统框架, 核心思想是多方数据拥有者参与联邦学习模型训练时, 不需要进行数据交互, 实现模型共建。即“数据不动, 模型动”的应用模式<sup>[18]</sup>。典型的联邦学习架构是由谷歌提出并建立的联邦平均算法 (FedAvg), 该架构由  $K$  个客户端和 1 个中央服务器组成。通常在每一轮的联邦学习中, 中央服务器都会从  $K$  个客户端中随机选取  $N$  个客户端参与联邦学习, 最终全局模型也是通过这  $N$  个客户端协同训练所得<sup>[19]</sup>。若每个客户端的本地数据集表示为  $\{x_i, y_i\} \in D$ , 其集合  $D$  的大小为  $D_i$ , 则对于客户端的损失函数定义为:

$$f_i(w) = l(w; x_i, y_i) \quad (1)$$

其中,  $w$  表示模型参数, 联邦学习的最终目的是找到使得损失函数最小化的最优模型参数  $w$ , 优化函数表示如下:

$$F(w) = \frac{1}{\sum_{i=1}^N D_i} \sum_{i=1}^N \sum_{\{x_i, y_i\} \in D_i} l(w; x_i, y_i) \quad (2)$$

在 FedAvg 算法中训练是由多轮通信组成。在每一轮通信中, 服务器选择一小部分节点,  $S_t = N = cK$  参加培训。以前一轮的全局模型  $w^{t-1}$  为参考, 每个参与节点  $i \in S_t$  执行局部随机梯度下降 (SGD) 算法更新权重。对此可以表示为:

$$w_i^t = w^{t-1} - \eta \nabla F_i(w^{t-1}) \quad (3)$$

其中,  $\eta$  表示学习率,  $\nabla F_i(\cdot)$  表示节点  $i$  处的梯度;  $w'_i$  是局部客户端更新后的结果。

式(3)给出了SGD优化的一般原理,然后参与节点将其模型更新  $\Delta'_i = w'_i - w^{t-1}$  返回给服务器,服务器对接收到的结果进行聚合,并进行全局模型更新。研究推得的公式可写为:

$$\Delta^t = \frac{1}{|S_t|} \sum_{i \in S_t} \Delta'_i \quad (4)$$

$$w^t = w^{t-1} + \Delta^t \quad (5)$$

联邦学习是一个迭代求解全局模型的过程,中央服务器协同本地客户端训练全局模型,直到全局模型收敛。

## 2.2 非独立同分布数据下的FedAvg算法

在谷歌提出的联邦学习FedAvg算法中,服务器对客户端的选取方式是从参与联邦学习的本地设备中随机选取一部分客户端进行联邦学习,在客户端数据分布较为平均的情况下,此算法在随机节点选择策略和简单的平均设计下可以达到很好的收敛速度,但是在部分非独立同分布数据客户端参与的情况下,FedAvg算法的收敛速度大幅下降<sup>[20]</sup>。这是因为局部优化器最小化的局部目标  $F_i(w)$  与数据分布密切相关。在FedAvg算法节点选择策略中,数据样本在所选节点上的分布彼此不同的情况下,局部更新导致模型朝向其局部目标的最优解,该局部目标在非独立同分布数据下偏离全局目标。最终使FL模型难以收敛。

基于FedAvg客户端选择算法在非独立同分布数据下存在的问题,研究从数据异构的角度理解和分析 non-iid 节点选择策略,认识到识别和选择更有利于模型收敛的节点是至关重要的。分析局部更新与全局更新之间的差异,进一步根据分析量化节点上的数据分布的差异,通过量化出来的差异识别出对全局更新有不利影响的节点。排除潜在的不利局部更新并降低这些节点被选择的概率,可以对具有更高贡献的节点分配更高的被选择概率。

## 3 基于节点贡献的联邦学习客户端选择算法

### 3.1 基于梯度信息的客户端列表聚合

考虑到上述问题,本文设计了一种基于 non-iid 的联邦学习客户端动态选择机制,首先初步筛选出有利于全局模型更新的局部客户端。通过对全局客户端进行检查,在每一轮联邦学习中,如果来自参与节点的局部更新,其梯度和全局梯度之间的内积为负数时 ( $\langle \nabla F(w^t), \nabla F_i(w^t) \rangle < 0$ ), 将会由于预

期损失减少而减缓全局模型的收敛速度。排除不利于全局更新的局部更新也会对联邦学习全局的数据异质性的降低产生影响。

为了找到有利于全局模型更新的局部更新,本文对局部更新和全局更新进行了期望计算  $\mathbb{E}_{i \sim S_t} [\langle \nabla F(w^t), \nabla F_i(w^t) \rangle]$ , 如果在当前轮检查出排除了某些客户端后满足  $\mathbb{E}_{i \sim \bar{S}_t} [\langle \nabla \bar{F}(w^t), \nabla F_i(w^t) \rangle] > \mathbb{E}_{i \sim S_t} [\langle \nabla F(w^t), \nabla F_i(w^t) \rangle]$ , 则排除  $S_t$  和  $\bar{S}_t$  两个集合的差集节点,即排除期望更低的客户端。但是排除局部更新会对全局更新和整体数据异构性产生影响,从而导致无法定量分析最后的结果,所以本文在第一步计算出期望之后选择采用计算测试损失。通过此方法不仅可以排除对全局模型训练不利的客户端,还可以在后续提高贡献更高客户端被选中的概率。

基于梯度信息的联邦学习客户端聚合算法的目的为通过排除不利局部更新,筛选出有利的客户端集合。算法1描述具体如下。

#### 算法1 基于梯度信息的联邦学习客户端聚合

**输入**  $S_t$  为参与联邦学习的客户端,  $\Delta'_i$  是局部客户端模型更新参数, num 为客户端阈值, dic 为带有键值对的字典

**输出** 全局模型  $w^{t+1}$

梯度聚合 (Gradient aggregation):

1. 初始化  $w_0, \nabla F_i(w^t) \leftarrow \frac{-\Delta'_i}{\eta}$
2. 全局梯度  $\nabla F(w^t) \leftarrow \frac{1}{|S_t|} \sum_{i \in S_t} \nabla F_i(w^t)$
3. 期望值初始化  $\max = \mathbb{E}_{i \sim S_t} [\langle \nabla F(w^t), \nabla F_i(w^t) \rangle]$
4. while  $S_t > \text{num}$  do
5.   期望值计算  $\text{dic} \leftarrow \text{Expectation}(\nabla F_i(w^t), S_t, \text{dic})$
6.   if 期望  $\text{dic} < \max$  then
7.     联邦学习客户端  $S^* = S_t$
8.   else
9.     客户端  $\text{id} = \max(\text{dic}). \text{key}$
10.    剩余客户端  $\bar{S}_t \leftarrow \text{del } S_t[\text{id}]$
11.     $w^{t+1} \leftarrow \nabla F_i(w^t) \quad i \in S_t$
12.     $\bar{w}^{t+1} \leftarrow \nabla F_i(w^t) \quad i \in \bar{S}_t$
13.    通过测试数据分布计算  $\text{loss}(w), \text{loss}(\bar{w})$

14. if  $\text{loss}(w) > \text{loss}(\bar{w})$  then
15.     剩余客户端  $S^* = S_i$
16. else
17.      $S_i, S^* \leftarrow \text{del } S_i[\text{id}]$
18.      $\text{max} = \text{dic}(\text{id}). \text{value}$
19.     end if
20. end if
21. end while
22.  $w^{t+1} \leftarrow$  式(3)和式(5)

期望值计算(Expected value calculation):

Expection ( $\nabla F_i(\cdot), S_i, \text{dic}$ )

1. for  $i = 1, 2, \dots$  do
2.     剩余客户端  $\bar{S}_i \leftarrow \text{del } S_i[i]$
3.      $\nabla F(w^t) \leftarrow \frac{1}{|\bar{S}_i|} \sum_{i \in \bar{S}_i} \nabla F_i(w^t)$
4.      $\text{dic}(S_i[i]) \leftarrow \mathbb{E}_{i \in \bar{S}_i} [\langle \nabla F(w^t), \nabla F_i(w^t) \rangle]$
5. end for

对于联邦学习的每一轮训练中通过服务器迭代对比计算 Expection 函数,如果在排除了某个局部更新后给出了更高的期望,则保留此客户端。

### 3.2 基于动态概率的联邦学习客户端选择

在联邦学习的每一轮训练中,都会选择一些节点参与局部训练和全局聚合。对于那些局部更新会减缓全局模型收敛的节点,算法需要降低这些节点被选择的概率。这里给出算法 2 描述具体如下。

**算法 2 基于动态概率的联邦学习客户端选择**

**输入**  $P_i$  为客户端概率,  $K$  为客户端数量,  $E$  是局部训练轮次,  $T$  是全局轮次

**输出** 全局模型  $w^{t+1}$

概率选择(Probabilistic selection):

1. 初始化概率  $P_i = \frac{1}{K}$
2. for  $t = 1, 2, \dots, T$  do
3.     服务器发送全局模型  $w^t$
4.     本地客户端用本地数据训练并发送  $\Delta_i^t$
5.      $w^{t+1} \leftarrow$  Gradient Aggregation
6.     根据 MargContribution 计算出的贡献与式(5)计算得到的概率  $P_i$
7.     return  $w$
8. end for

贡献值计算(Contribution calculation):

MargContribution:

1. for  $i = 1, 2, \dots, K$  do

2.     if  $i$  不在剔除队列 then
3.          $\text{coalition}_i \leftarrow S_i[:i] + S_i[i+1:]$
4.          $\text{marginal} \leftarrow$  合作价值函数  $\text{cooperative}(S_i) - \text{cooperative}(\text{coalition}_i)$
5.          $\text{contribution}[i] \leftarrow \frac{\text{marginal}}{K}$
6.     end if
7. end for
8. return contribution

因此,在服务器端,本文基于算法 2 得到的客户端队列根据之后计算得到的客户端的贡献动态地改变每个节点被选择的概率。由算法 2 剔除的那些节点的概率将减少,而其余的节点通过每个节点的边际贡献值获得相应的增加。被标记的节点概率在本轮训练中将被减少为 0。在标记的节点概率变化之后,其余剩下的节点概率将会得到增加,具体的概率增加需要结合每个节点对于全局模型的贡献,具体概率公式如下:

$$P_i^{t+1} = P_i^t + \Delta P_i^t \times \frac{\text{con}[i]}{\text{sum}(\text{con})} \quad (6)$$

其中,  $P_i^t$  表示第  $i$  个节点在第  $t$  轮下被选择概率,  $\Delta P_i^t$  表示第  $i$  个节点在第  $t$  轮下被改变的的概率,每个节点获得的概率由该节点的贡献占比决定。在 MargContribution 函数中,本文采用损失函数作为模型的评估函数来计算贡献值。

本文提出的改进方法 FedDPCS 算法的流程如图 1 所示。图 1 左边为客户端群体,右边为服务器,在每一联邦学习轮中,客户端发送更新后的模型参数到服务器,服务器首先用算法 1 项排除不利的局部更新,第二步根据 MargContribution 函数计算客户端的贡献值,最后服务器根据算法 2 计算出客户端的概率变化,由服务器根据最终概率选择客户端进行模型聚合,下发全局模型参数,重复以上步骤,直至模型收敛。

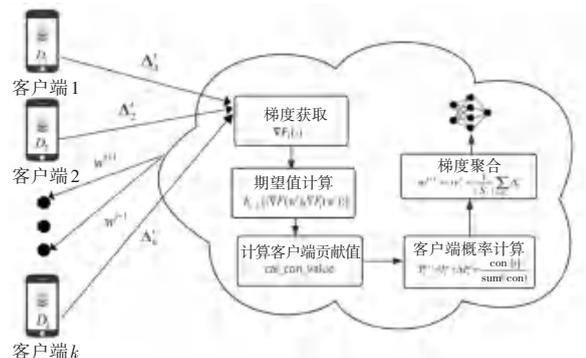


图 1 FedDPCS 算法流程

Fig. 1 Process of FedDPCS algorithm

## 4 实验与结果分析

### 4.1 实验设置

为了评估本文提出的基于非独立同分布的联邦学习客户端选择方法,在 Pytorch 框架下模拟中央服务器与客户端进行联邦学习的过程,分别在数据集 MNIST<sup>[21]</sup> 上进行了实验。

首先本文用 MNIST 数据集在 MLR 模型上进行了凸分类问题训练,然后在 CNN 模型上用 MNIST 和 CIFAR-10 数据进行了非凸分类问题训练。MNIST 数据集为 70 000 张  $28 \times 28$  的灰度图像,共 10 个类别,每个类别代表 0~9 之间的一个数字,每张图像只有一个类别。训练集 60 000 张图像,测试集 1 000 张图像,CIFAR-10 是一个图像分类数据集,其中包含 10 类不同类型的彩色图片,每类图片有 6 000 张,共计 60 000 张图片。每张图片的分辨率为  $32 \times 32$  像素,包含 3 个颜色通道(RGB),因此每张图片的大小为  $32 \times 32 \times 3$ 。

在实验中模拟生成客户端数量为 50,用于节点选择比例参数为  $c = 0.2$ ,  $S_i = N = cK = 10$ ,学习率  $\eta = 0.01$ ,在 MNIST 和 CIFAR-10 数据集上进行实验时,为每个客户端分配的数据样本数量为 200,对于

数据异质性的设置,本文主要由非独立同分布数据节点的占比  $\alpha$  和节点数据的偏度  $\beta$  来衡量。例如  $\alpha = 0.1$ ,表示只有 5 个客户端上的数据是独立同分布的,节点数据的偏度  $\beta$  则表示客户端上数据的方差,方差越小代表数据异质性越高,为使实验结果简单明了,本文设置了  $\alpha$  参数为 0.2 和 0.5,  $\beta$  参数为 1 和 2。

### 4.2 实验结果与分析

(1) 实验 1,在 MNIST 数据集上训练 MLR 模型和 CNN 模型。MLR 模型和 CNN 模型在 MNIST 数据集上的实验结果如图 2 所示。从图 2 中可以看出,与 MLR 模型和 CNN 模型下的 FedAvg 和 FedBn2<sup>[23]</sup> 相比,无论数据异构性的程度如何, FedDPCS 算法收敛速度更快,测试精准度更高。与训练学习能力较弱的 MLR 模型的情况相比,采用 CNN 模型时本文算法获得了更好更明显的改善。此外,还能观察到随着数据异质性变得更高,本文算法的性能表现也更加优异,这也验证了本文算法的有效性;对于数据异构性较低的情况,究其原因就在于此时会选择大量的拥有独立同分布数据的客户端,所以导致效果并不明显。

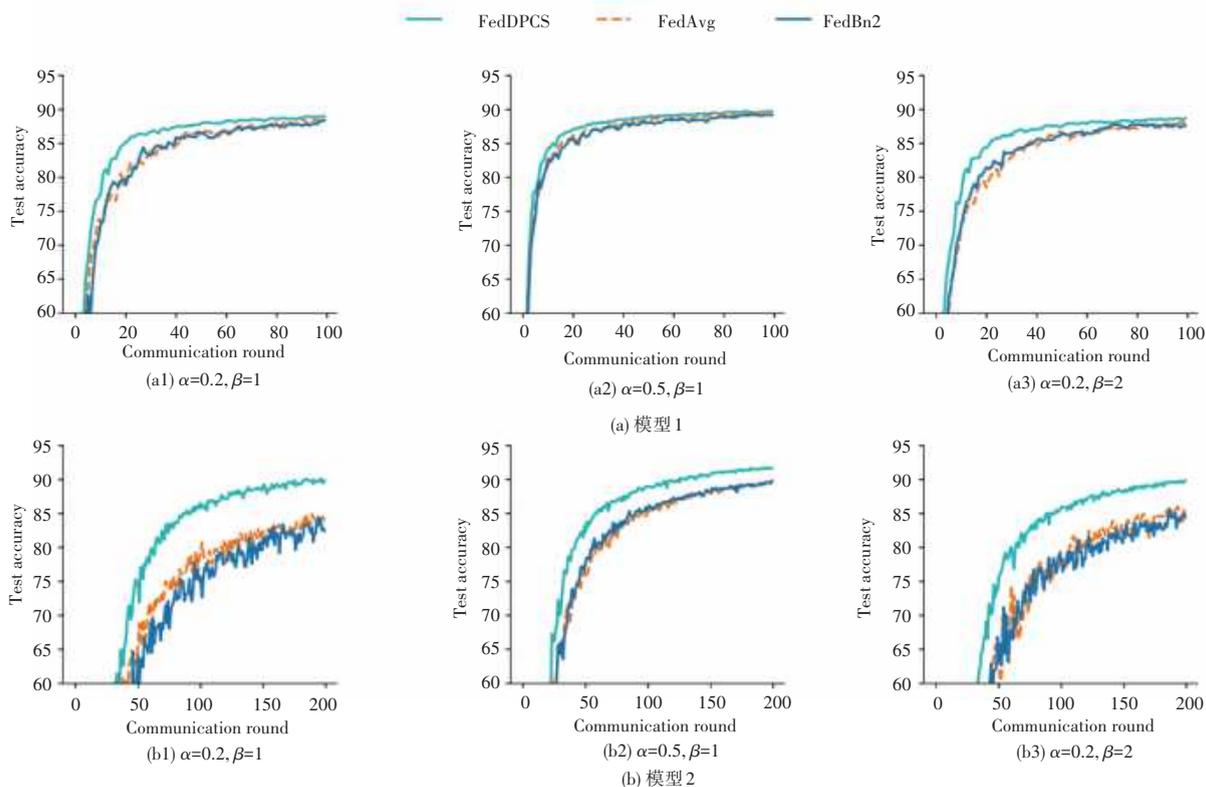


图2 MLR模型和CNN模型在MNIST数据集上的实验结果

Fig. 2 The experimental results of the MLR and CNN models on the MNIST dataset

(2) 实验 2, 在 CIFAR-10 数据集上训练 CNN 模型。对于更加复杂的三通道图像数据集, 本实验中设置客户端本地训练轮次是 5 轮, 研究得到 CNN 模型在 CIFAR-10 数据集上的实验结果如图 3 所

示。从图 3 中可以看出, 与 FedAvg 和 FedBn2 相比, FedDPCS 算法有着更快的收敛速度和更高的测试精度、特别是在数据异质性较高的情况下, 这也证明了本文算法有效性。

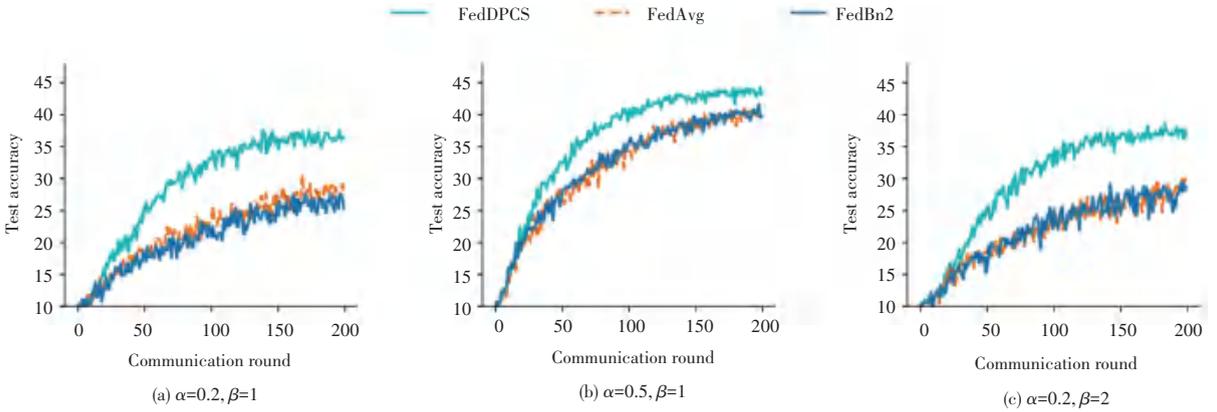


图 3 CNN 模型在 CIFAR-10 数据集上的实验结果

Fig. 3 Experimental results of the CNN model on the CIFAR-10 dataset

## 5 结束语

本文从联邦学习环境下客户端数据异构性角度出发, 从联邦学习客户端的选择入手对联邦学习效率进行优化, 提出了面向非独立同分布的联邦学习客户端选择方法, 即 FedDPCS 算法。该算法首先通过客户端的局部梯度以及全局梯度排除不利于全局的客户端, 其次根据贡献算法, 计算剩余的客户端贡献, 通过贡献分配概率, 服务器通过最终概率选择客户端。最后, 在不同的模型、数据集上的实验结果表明, 与 FedAvg 以及 FedBn2 算法相比, 改进算法 FedDPCS 在进行 FL 训练时, 对模型的收敛速度以及模型的精度都有所提高。

## 参考文献

- [1] 周传鑫, 孙奕, 汪德刚, 等. 联邦学习研究综述 [J]. 网络与信息安全学报, 2021, 7(5): 77-92.
- [2] 微众银行 AI 项目组. 联邦学习白皮书 V1. 0 [R]. 深圳: WeBank, 2018.
- [3] MCMAHAN B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data [J]. arXiv preprint arXiv, 1602.05629, 2023.
- [4] LI Tian, SAHU A K, ZAHEER M, et al. Federated optimization in heterogeneous networks [J]. arXiv preprint arXiv, 1812.06127, 2018.
- [5] WANG Hao, KAPLAN Z, NIU Di, et al. Optimizing federated learning on non-iid data with reinforcement learning [C]// Proceedings of the IEEE Conference on Computer Communications. Piscataway, NJ: IEEE, 2020:1-10.
- [6] ZHAO Y, LI M, LAI L, et al. Federated learning with non-IID

- data [J]. arXiv preprint arXiv, 1806.00582, 2018.
- [7] NAGALAPATTI L, NARAYANAM R. Game of gradients: Mitigating irrelevant clients in federated learning [C]// Proceedings of the AAAI Conference on Artificial Intelligence. AAAI, 2021: 9046-9054.
- [8] KAIROUZ P, MCMAHAN H B, AVENT B, et al. Advances and open problems in federated learning [J]. arXiv preprint arXiv, 1912.04977, 2019.
- [9] WANG Shiqiang, TUOR T, SALONIDIS T, et al. Adaptive federated learning in resource constrained edge computing systems [J]. IEEE Journal on Selected Areas in Communications, 2019, 37(6): 1205-1221.
- [10] WANG Luping, WANG Wei, LI Bo. CMFL: Mitigating communication overhead for federated learning [C]// Proceedings of 2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS). Piscataway, NJ: IEEE, 2019: 954-964.
- [11] LIU Lumin, ZHANG Jun, SONG S, et al. Client-edge-cloud hierarchical federated learning [C]// Proceedings of 2020 IEEE International Conference on Communications (ICC). Piscataway, NJ: IEEE, 2020:1-6.
- [12] WU Hongda, WANG Ping. Fast-convergent federated learning with adaptive weighting [J]. IEEE Transactions on Cognitive Communications and Networking, 2021, 7(4): 1078-1088.
- [13] NAGALAPATTI L, MITTAL R S, NARAYANAM R. Is your data relevant? Dynamic selection of relevant data for federated learning [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(7): 7859-7867.
- [14] CHO Y J, WANG Jianyu, JOSHI G J A P A. Client selection in federated learning: Convergence analysis and power-of-choice selection strategies [J]. arXiv preprint arXiv, 2010.01243, 2020.
- [15] REN Jinke, HE Yinghui, WEN Dingzhu, et al. Scheduling for cellular federated edge learning with importance and channel awareness [J]. IEEE Transactions on Wireless Communications, 2020, 19(11): 7690-7703.

- [16] RIZK E, VLASKI S, SAYED A H. Optimal importance sampling for federated learning [C]// Proceedings of 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway, NJ: IEEE, 2021: 3095 – 3099.
- [17] ZHANG Yufeng, LI Peng, WANG Kun, et al. Big data analytics by crowdlearning: Architecture and mechanism design [J]. IEEE Network, 2020, 34(3): 143–147.
- [18] SOLTANI B, HAGHIGHI V, MAHMOOD A, et al. A survey on participant selection for federated learning in mobile networks [J]. arXiv preprint arXiv, 2207.03681, 2022.
- [19] 何常乐, 袁培燕. 边缘联邦学习的客户端选择机制 [J]. 计算机应用, 2023, 43(S1): 147–153.
- [20] LI Xiang, HUANG Kaixuan, YANG Wenhao, et al. On the convergence of fedavg on non – IID data [J]. arXiv preprint arXiv, 1907.02189, 2019.
- [21] LECUN Y, CORTES C, BURGES C. MNIST handwritten digit database [EB/OL]. (2010–05–01). <http://yann.lecun.com/exdb/mnist/>.
- [22] KRIZHEVSKY A, NAIR V, HINTON G J. Cifar–10 (canadian institute for advanced research) [EB/OL]. (2010–05–01). <http://www.cs.toronto.edu/kriz/cifar>.
- [23] AMIRI M M, GÜNDÜZ D, KULKARNI S R, et al. Convergence of update aware device scheduling for federated learning at the wireless edge [J]. arXiv preprint arXiv, 2001.10402, 2020.