计算机科学
COMPUTER SCIENCE

面向长尾异构数据的个性化联邦学习框架

吴家皋, 易婧, 周泽辉, 刘林峰

引用本文

吴家皋, 易婧, 周泽辉, 刘林峰. 面向长尾异构数据的个性化联邦学习框架[J]. 计算机科学, 2025, 52(9): 232-240.

WU Jiagao, YI Jing, ZHOU Zehui, LIU Linfeng. Personalized Federated Learning Framework for Long-tailed Heterogeneous Data [J]. Computer Science, 2025, 52(9): 232-240.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.240700116

面向长尾异构数据的个性化联邦学习框架

吴家皋　易　婧　周泽辉　刘林峰
南京邮电大学计算机学院　南京 210023
江苏省大数据安全与智能处理重点实验室　南京 210023

摘　要　针对数据长尾分布和异构性引起的联邦学习模型性能下降的问题,提出了一种新的个性化联邦学习框架——平衡的个性化联邦学习(Balanced Personalized Federated Learning,BPFed),将整个联邦学习过程分为基于个性化联邦学习的表示学习和基于全局特征增强的个性化分类器再训练两个阶段。在第一阶段,首先采用 Mixup 策略进行数据增强,然后提出基于参数解耦的个性化联邦学习特征提取器训练方法,在优化特征提取器性能的同时减少通信开销;在第二阶段,首先提出新的基于全局协方差矩阵的类级特征增强方法,然后提出基于样本权重的标签平滑损失函数对客户端分类器进行平衡的个性化再训练,以纠正头类置信过度并提高尾类的泛化能力。大量的实验结果表明,在不同的数据长尾分布和异构性设置下,BPFed 模型的准确度相比其他代表性相关算法均有明显提升。此外,消融和超参数影响实验也进一步验证了所提方法和优化策略的有效性。

关键词:个性化联邦学习;长尾分布;数据异构性;参数解耦;特征增强;优化策略
中图分类号　TP393

Personalized Federated Learning Framework for Long-tailed Heterogeneous Data

WU Jiagao,YI Jing,ZHOU Zehui and LIU Linfeng
School of Computer,Nanjing University of Posts and Telecommunications,Nanjing 210023,China
Jiangsu Key Laboratory of Big Data Security & Intelligent Processing,Nanjing 210023,China

Abstract　Aiming at the problem of model performance degradation in federated learning caused by long tail distribution and heterogeneity of data,a novel personalized federated learning framework called Balanced Personalized Federated Learning(BPFed) is proposed,where the whole federated learning process is divided into two stages:representation learning based on personalized federated learning and personalized classifiers retraining based on global feature augmentation. In the first stage,the Mixup strategy is adopted firstly for data augmentation,and then a feature extractor training method is proposed based on the personalized federated learning with parameter decoupling to optimize the performance of the feature extractor while reducing the communication cost. In the second stage,a new class-level feature augmentation method based on global covariance matrix is proposed firstly,and then the classifiers of clientsare retaines individually in balance with a proposed label-aware smoothing loss function based on sample weight to correct the overconfidence for head classes and boost the generalization ability for tail classes. Extensive experimental results show that the model accuracy of BPFed is significantly improved compared with other representative related algorithms under different settings of data long-tailed distributions and heterogeneities. Moreover,the effectiveness of the proposed methods and optimization strategies is further verified by the experiments on ablation and hyperparameter influence.

Keywords　Personalized federated learning,Long-tailed distribution,Data heterogeneity,Parameter decoupling,Feature augmentation,Optimization strategy

1　引言

在信息化快速发展的今天,数据已成为推动人工智能和机器学习进步的关键资源[1]。传统的机器学习方法通常采用集中式训练模式,即将数据集中到服务器上进行模型训练。然而,这种方式在处理敏感数据时可能带来安全威胁和隐私泄露的风险。随着人们对数据安全的日益重视,集中式训练显然已不再适用。

联邦学习[2](Federated Learning,FL)是一种新兴的分布式机器学习框架,其允许多个参与方在不共享本地数据的前提下,通过本地模型的聚合共同训练全局模型,有效避免了数据泄露的风险,为数据隐私保护提供了一种有效的解决方案。然而,FL 面临着一个重大挑战,即由参与设备(客户端)的地理位置、用户偏好等因素的差异所导致的数据异构性问题。

到稿日期:2024-07-18　返修日期:2024-10-17
基金项目:国家自然科学基金(62272237,61872191)
This work was supported by the National Natural Science Foundation of China(62272237,61872191).
通信作者:吴家皋(jgwu@njupt.edu.cn)

数据异构性往往使得各客户端的本地数据呈非独立同分布(Non-Independent and Identically Distributed，non-IID)性质，这将为全局模型聚合带来不可预测的偏差，从而降低 FL 模型的准确度和收敛速度。此外，在现实世界中，训练数据样本通常表现为长尾(Long-tailed)分布[3]特征，即少数类别(头类)拥有大量样本，而多数类别(尾类)样本稀少。长尾分布特征将导致训练出的模型倾向于优化样本量大的头类的性能，而忽略了样本量少的尾类。同时，在 FL 环境下，尾类样本可能仅在少数客户端上零星存在，这进一步加剧了模型在尾类上性能不足的问题。

目前，研究者针对长尾学习问题已经提出了多种解决策略，主要可分为 3 个方面：重采样/重加权、数据/特征增强和模型改进。重采样/重加权通过调整采样频率[4]或损失权重[5]来平衡类别分布；数据/特征增强通过生成新样本[6]或改进特征表示[7]来提高少数类的识别能力；模型改进通过改善网络结构和训练策略来校正失衡的分类器。其中，解耦训练[4]是模型改进的主要策略之一，该策略将模型训练分为两阶段，通过先训练特征提取器，再优化分类器来提升模型性能。然而，上述集中式方法在 FL 环境中将面临数据异构性和隐私保护等新的挑战。近年来，基于 FL 的长尾问题研究已取得一定成果[8-9]，但这些工作仍存在需共享信息和增加通信开销等问题。特别是，在异构 FL 环境下，训练单一全局模型显然无法保证其对每个客户端数据分布的泛化性能。为此，研究人员又进一步提出了个性化联邦学习(Personalized Federated Learning，PFL)方法。PFL 通过为每个客户端定制个性化的本地模型，能够更好地适应客户端各自的数据分布，从而提升模型在每个客户端上的性能。例如，FedPer[10]通过参数解耦将模型分解为基础层和个性化层，在训练时仅共享基础层参数，每个客户端的个性化层参数保留在本地，从而实现个性化模型训练。目前已有基于 PFL 的长尾学习的研究[11]，然而，这方面的工作仍处于初级阶段，很多问题有待解决。

因此，探索能同时处理数据 non-IID 和 long-tailed 分布特性的有效方法，以提高 FL 模型的整体性能和泛化能力，成为相关研究领域亟待解决的重要课题。为此，面向长尾异构数据，本文提出了一种新的 PFL 框架，称为平衡的个性化联邦学习(Balanced Personalized Federated Learning，BPFed)。该框架仍采用现有的两阶段训练方法，旨在将参数解耦 PFL[10]与解耦训练的长尾学习方法[4]有机结合。同时，在每个阶段综合运用多种策略进一步优化其性能。

本文的主要贡献如下：

1)对长尾异构数据分布条件下的 PFL 问题及优化目标进行了形式化描述，并提出了一种新的 PFL 框架——BPFed，通过基于 PFL 的表示学习和基于全局特征增强的个性化分类器再训练两个阶段，分别解决数据异构性和长尾分布不平衡问题，从而有效增强客户端本地尾类的性能，提高个性化模型的整体性能。

2)综合运用多种策略提高各阶段的性能，在第一阶段，采用了 Mixup 数据增强策略，并提出了基于参数解耦的 PFL 特征提取器训练方法，在减少通信开销的同时得到性能更优的特征提取器；在第二阶段，提出了新的基于全局协方差矩阵的类级特征增强方法和基于样本权重的标签平滑损失函数，以实现平衡的个性化分类器再训练，在提高尾类泛化性的同时有效纠正头类置信过度问题。

3)不同数据长尾分布和异构性设置下的实验结果表明，与现有方法相比，BPFed 显著提高了每个客户端本地个性化模型的性能。

2 相关工作

2.1 长尾学习

在许多实际问题中，训练样本通常表现为长尾类分布，即其中一小部分头类具有大量的样本，而其他大部分尾类只有少数样本。在此情况下，训练后的模型很容易偏向头类，而在尾类上表现不佳[3]。为解决长尾类失衡问题，近年来开展了大量的长尾学习研究。

目前，在集中式长尾学习方面已有较为成熟的解决方案，包括重采样/重加权、数据/特征增强和模型改进等策略。重采样和重加权通过调整采样频率[4]或损失权重[5]来平衡类别分布，缓解模型对头类的偏好。数据/特征增强技术通过变换或扩展尾类样本[6]，提升模型对这些类别的学习能力。此外，基于模型改进的解耦训练策略将训练过程分为两个阶段[4]：第一阶段通过实例平衡采样来学习数据的表示，第二阶段则固定表示参数，通过类平衡采样重新训练分类器。该方法能提升模型对尾类的识别准确率，对长尾分布数据具有较高的泛化能力。Zhong 等[12]进一步探索了长尾识别中的校准模型方法，通过 Mixup 策略增强了模型对不同类别的适应能力，并通过标签感知平滑技术调整标签的置信度来减少模型对数据不平衡的敏感性。然而，上述策略在 FL 环境中直接应用会受到隐私保护和数据异构性的双重挑战。例如，重采样以及数据/特征增强技术可能需要跨节点共享信息以实现适当的样本平衡，但这可能会泄露敏感数据。重加权虽然可以提升分类器对尾类的性能，但可能导致头类精度显著下降。而模型改进的方法也需依赖于对数据的访问，FL 服务器同样无法通过直接访问数据来执行必要的类平衡采样或再训练过程。在 FL 领域，近来也有一些针对长尾问题的研究工作。基于解耦训练策略，CReFF[8]同样采用了两阶段方法：在第一阶段，系统进行常规的 FL 训练；在第二阶段，系统通过聚合来自各客户端的每类联邦特征梯度来生成联邦特征，并利用这些联邦特征重新训练分类器，以提升模型的整体表现。但这种方法也增加了通信成本和隐私泄露的风险。Balance-FL[9]通过在本地训练前进行知识继承和特征增强来应对数据不平衡问题，但其有效性因设备间数据统计的差异而受到限制。在中文文献方面，Wang 等[13]提出了一种基于组合式多臂老虎机框架的 FL 设备选择算法。通过类别估计方案，选取类别测试性能最互补的设备子集参与每轮训练，从而获得性能更平衡的全局模型。Wu 等[14]提出了一种面向全局不平衡问题的 FL 方法，设计了基于贡献度的全局判别损失函数和模型加权聚合策略，以提高模型对尾类的泛化能力。但上述两种方法都要求在服务器端保留少量全局共享的测试数据集，这可能导致隐私保护问题，进而影响其应用范围。

2.2 个性化联邦学习

在 FL 中，客户端数据的异构性，即 non-IID 分布，是引发全局模型偏离，从而影响 FL 训练收敛与性能的主要原因。为应对数据异构性挑战，目前已提出多种优化策略[15-21]。这些策略主要致力于增强全局模型的稳定性和泛化能力，以适应客户端之间的差异。然而，单一的全局模型通常无法很好地保证对不同的客户端数据分布都有良好的泛化性能，特别是在异构性较高的情况下。幸运的是，PFL 的提出为此提供了一种新的思路，其目标是为不同客户端生成适应其数据集的定制化模型。FedPer[10]是 PFL 中的典型框架，它将模型参数分解为基础层与个性化层。基础层负责捕捉全局数据特征并通过服务器聚合以及在客户端之间共享，而个性化层则留在各个客户端本地，专注于学习每个客户端特有的数据特征，从而更好地适应各客户端的数据分布。在此基础上，Liu 等[22]提出了一种基于参数解耦和聚类的 PFL 算法，用以实现对工业物联网设备资源的高效调度。

但 FedPer 在处理长尾分布的数据时仍然存在长尾类失衡问题。Lu 等[11]提出的 FedAFA 为解决个性化长尾问题提供了一种新方法，它在传统 FL 基础上，通过生成对抗性特征在第二阶段对分类器进行个性化重训练，以缓解长尾分布和数据异构性带来的不利影响。然而，FedAFA 在第一阶段并未采用 PFL 方式，也未对特征提取和分类器再训练进行优化，因此仍有较大的改进空间。

针对现有工作的不足，本文提出了 BPFed 框架，在保护数据隐私的同时，综合运用多种策略，有效解决了数据异构性和长尾失衡问题。

3 系统模型

3.1 问题描述

1)本地和全局数据分布：本文考虑一个具有 K 个客户端的个性化 FL 系统和一个用于分类问题的 C 类视觉识别数据集。设 $\boldsymbol{D}_k=\{(\boldsymbol{x}_i,\boldsymbol{y}_i)\}_{i=1}^{n_k}$ 表示客户端 $k\in[1,K]$ 的本地私有数据集，其中，$\boldsymbol{x}_i\boldsymbol{D}_k$ 中的第 i 个样本，$\boldsymbol{y}_i=[y_{i,1},\cdots,y_{i,C}]\in\mathbb{R}^C$ 为第 i 个样本独热编码的真实标签。这里，$y_{i,c}=1$ 表明 $\boldsymbol{x}_i$ 属于第 c 类，而所有其他 $y_{i,j}=0(\forall j\neq c)$。$n_k$ 表示客户端 k 的本地数据集大小(即$|\boldsymbol{D}_k|$)。令 n_k^c 表示 $\boldsymbol{D}_k$ 中第 c 类数据样本的数量，则 $n_k=\sum_{c=1}^{C}n_k^c$。对于给定的客户端 k，其本地数据分布 $\mathbb{S}_k\in\mathbb{R}^C$ 可表示为：

$$\mathbb{S}_k=\{n_k^1,\cdots,n_k^C\} \tag{1}$$

为了更好地从系统层面描述整体数据分布，全局数据分布 $\mathbb{S}_G\in\mathbb{R}^C$ 可定义为系统中所有客户端数据集的聚合分布。

$$\mathbb{S}_G=\{\sum_{k=1}^{K}n_k^1,\cdots,\sum_{k=1}^{K}n_k^C\} \tag{2}$$

对于长尾分布的全局数据集，令类别的样本数量非递增排序，则 $\forall c_1<c_2$，有 $\sum_{k=1}^{K}n_k^{c1}\geqslant\sum_{k=1}^{K}n_k^{c2}$，并且 $\sum_{k=1}^{K}n_k^1\gg\sum_{k=1}^{K}n_k^C$。

2)分类模型：不失一般性地，设分类模型为 $\phi_{\boldsymbol{W}}$，其中，$\boldsymbol{W}$ 表示该模型的参数。通常模型 $\phi_{\boldsymbol{W}}$ 又可分解成两个主要组成部分，即 $\phi_{\boldsymbol{W}}=\{f_U,h_V\}$，其中，$f_U$ 表示参数为 $\boldsymbol{U}$ 的特征提取器，h_V 表示参数为 $\boldsymbol{V}$ 的分类器，$\boldsymbol{W}=\{\boldsymbol{U},\boldsymbol{V}\}$。令 $(\boldsymbol{x},\boldsymbol{y})$ 为任一视觉识别数据样本/标签对，特征提取器 f_U 负责将输入样本 $\boldsymbol{x}$ 映射至一个 d 维的特征向量 $\boldsymbol{z}$，即 $\boldsymbol{z}=f_U(\boldsymbol{x})\in\mathbb{R}^d$；分类器 h_V 通常是一个全连接层，对给定特征向量 $\boldsymbol{z}$ 进行回归得到一个概率向量 $\boldsymbol{p}=h_V(\boldsymbol{z})\in\mathbb{R}^C$，令 $\boldsymbol{p}=[p_1,\cdots,p_C]$，该向量中的每一个元素 p_c 代表样本属于对应类别 c 的概率。具体的预测类别 $\hat{c}$ 可以表示为：

$$\hat{c}=\arg\max_{c\in[1,C]}\{p_c\} \tag{3}$$

在 PFL 中，允许每个客户端学习训练不同的本地模型参数。因此，对于客户端 $k\in[1,K]$，其本地模型的参数可记作 $\boldsymbol{W}_k=\{\boldsymbol{U}_k,\boldsymbol{V}_k\}$。

3)优化目标：在 non-IID 的情况下，各客户端的本地数据集可能并不拥有所有类别的样本。针对这一问题，引入拥有类与缺失类的概念[9]，令 $\boldsymbol{C}_k^{\text{pos}}$ 和 $\boldsymbol{C}_k^{\text{neg}}$ 分别表示客户端 k 拥有的和缺失的样本类别集合，且满足 $C=|\boldsymbol{C}_k^{\text{pos}}|+|\boldsymbol{C}_k^{\text{neg}}|$。BPFed 的最终优化目标是通过 PFL 获得 K 个客户端的个性化模型 $\phi_{\boldsymbol{W}_k}$，以在其客户端拥有类 $\boldsymbol{C}_k^{\text{pos}}$ 中实现最佳的整体预测精度。其中，客户端拥有的每个类别都应该视为同等重要。

具体而言，训练过程基于分布于各个客户端的数据进行，这些数据遵循各自的局部分布 $\{\mathbb{S}_k\}_{k=1}^{K}$ 以及一个全局分布 $\mathbb{S}_G$。在测试阶段，最终目标是最大化所有客户端拥有类的平均测试准确度。对于客户端 k，其使用个性化模型 $\phi_{\boldsymbol{W}_k}$ 在拥有类 $\boldsymbol{C}_k^{\text{pos}}$ 上的平均测试准确度 $acc_k(\phi_{\boldsymbol{W}_k})$ 可表示为：

$$acc_k(\phi_{\boldsymbol{W}_k})=\sum_{c\in\boldsymbol{C}_k^{\text{pos}}}\frac{acc_k^c(\phi_{\boldsymbol{W}_k})}{|\boldsymbol{C}_k^{\text{pos}}|} \tag{4}$$

其中，$acc_k^c(\phi_{\boldsymbol{W}_k})$ 表示客户端 k 使用个性化模型 $\phi_{\boldsymbol{W}_k}$ 在第 c 类样本上的测试准确度，其计算式如下：

$$acc_k^c(\phi_{\boldsymbol{W}_k})=\frac{\sum_{\boldsymbol{x}\in\{\boldsymbol{x}|(\boldsymbol{x},\boldsymbol{y})\in\boldsymbol{D}_k^c\}}\mathbb{I}(\hat{c}_k(\boldsymbol{x})=c)}{n_k^c} \tag{5}$$

其中，$\hat{c}_k(\boldsymbol{x})$ 为模型 $\phi_{\boldsymbol{W}_k}$ 对样本 $\boldsymbol{x}$ 的预测类别，具体结果由式(3)给出；$\mathbb{I}(\cdot)$ 为指示函数，当条件成立时值为 1，否则为 0；$\boldsymbol{D}_k^c$ 表示客户端 k 上真实类别为 c 的数据子集。

因此，异构长尾 PFL 的优化目标可以表示为：

$$\max_{\boldsymbol{W}_1,\cdots,\boldsymbol{W}_K}\frac{1}{K}\sum_{k=1}^{K}acc_k(\phi_{\boldsymbol{W}_k}) \tag{6}$$

3.2 BPFed 框架

为了解决数据长尾分布以及异构性的 FL 联合问题，本文提出了一种新型的长尾个性化 FL 框架 BPFed。图 1 展示了 BPFed 的总体框架，分为两个主要阶段。第一阶段基于 PFL 的表示学习阶段，第二阶段基于全局特征增强的个性化分类器再训练阶段。在第一阶段，采用 Mixup 策略对客户端的私有数据集进行数据样本增强。然后，通过基于参数解耦的 PFL 训练优化全局特征提取器。在第二阶段，每个客户端先利用第一阶段训练得到的全局特征提取器提取特征，得到本地协方差矩阵并上传。服务器利用客户端上传的协方差矩阵计算全局协方差矩阵并下发。然后，每个客户端基于全局协方差矩阵的类级特征增强方法对每个客户端的特征空间进行平衡处理。最后，在平衡特征空间的基础上，每个客户端通过基于样本权重的标签平滑损失函数对个性化分类器进行再训练，进一步提升尾类泛化能力和分类性能。

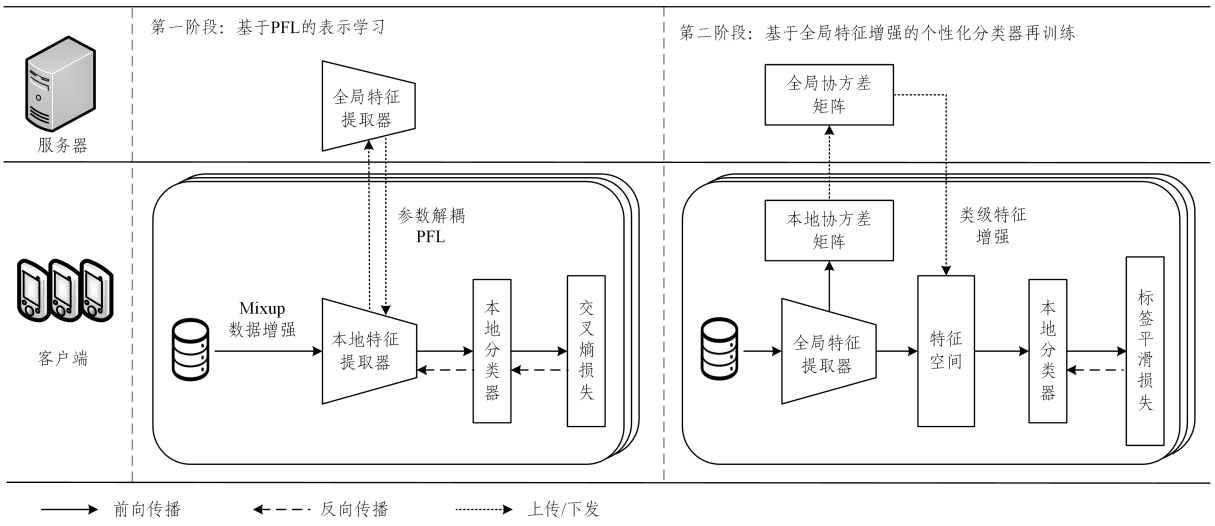


图 1 BPFed 框架

Fig. 1 Framework of BPFed

3.2.1 基于 PFL 的表示学习阶段

1)Mixup 数据增强

Mixup[6]是一种数据增强方法,通过在训练过程中将两个不同样本的特征和标签进行线性插值,生成新的训练样本,从而促进模型学习更具泛化性的特征表示。研究表明[12],在长尾分布的集中式训练时使用 Mixup 策略能够对表示学习产生积极影响。因此,在 BPFed 第一阶段引入这一策略来增强客户端的私有数据集,获得性能更优的特征提取器。

具体地,对于客户端 k 的本地私有数据集 $\boldsymbol{D}_k$,随机抽取两个样本 i 和 j 及其标签,即 $(\boldsymbol{x}_i,\boldsymbol{y}_i)$和$(\boldsymbol{x}_j,\boldsymbol{y}_j)$。则 Mixup 通过式(7)和式(8)生成新的样本/标签对$(\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})$:

$$\tilde{\boldsymbol{x}}=\lambda\boldsymbol{x}_i+(1-\lambda)\boldsymbol{x}_j \tag{7}$$

$$\tilde{\boldsymbol{y}}=\lambda\boldsymbol{y}_i+(1-\lambda)\boldsymbol{y}_j \tag{8}$$

其中,λ 为混合超参数,用于调节两个原始数据对新生成数据的影响。根据文献[6],超参数 λ 的取值由 Beta 分布 $Beta(\alpha,\alpha)$随机决定,α 的取值范围为$(0,\infty)$。关于 α 的具体取值及其应用细节,将在 4.3.3 节中详细讨论。

本文采用 Mixup 方法生成与原数据集 $\boldsymbol{D}_k$ 相同数量的新样本/标签对$(\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})$。所有这些新生成的样本/标签对构成新的数据集 $\widetilde{\boldsymbol{D}}_k$,用于后续的特征提取器训练。

2)基于参数解耦的 PFL 特征提取器训练

考虑到数据异构性问题以及参数解耦 PFL[10]与解耦训练的长尾学习[4]之间的方法相似性,本文提出基于参数解耦的 PFL 特征提取器训练方案,在此过程中,每个客户端仅需将其特征提取器参数上传到服务器并进行聚合,而其分类器则被保留在本地。采用此策略不仅减少了上传通信开销,更能在全局共享知识与本地个性化知识之间取得平衡,从而优化全局特征提取器。

设 T 为 PFL 全局训练的总轮数,t 为当前训练轮数,$\boldsymbol{W}_k^t=\{\boldsymbol{U}_k^t,\boldsymbol{V}_k^t\}$为第 t 轮客户端 k 的本地模型参数,$\boldsymbol{U}_{\text{glo}}^t$为第 t 轮服务器上的全局特征提取器参数。在 PFL 的初始阶段,即 $t=0$ 时,服务器和所有客户端都随机初始化各自的模型参数 $\boldsymbol{U}_{\text{glo}}^0$和 $\boldsymbol{W}_k^0$。第 t 轮训练过程如下:首先,服务器随机选择一组数量为$\lceil K\rho\rceil$的客户端集合 $\boldsymbol{A}^t$ 参与 FL,并将全局特征提取器参数 $\boldsymbol{U}_{\text{glo}}^t$分发给被选中的客户端,其中 ρ 为选择比例;随后,客户端 k 将其本地特征提取器参数 $\boldsymbol{U}_k^t$ 更新为全局特征提取器参数 $\boldsymbol{U}_{\text{glo}}^t$,即 $\boldsymbol{W}_k^t=\{\boldsymbol{U}_{\text{glo}}^t,\boldsymbol{V}_k^t\}$;接下来,客户端进行本地训练,按式(9)在其私有数据集 $\widetilde{\boldsymbol{D}}_k$ 上更新本地模型参数。

$$\boldsymbol{W}_k^{t+1}\leftarrow\boldsymbol{W}_k^t-\eta\nabla\ell_k(\boldsymbol{W}_k^t;\widetilde{\boldsymbol{D}}_k) \tag{9}$$

其中,$\ell_k(\boldsymbol{W}_k^t;\widetilde{\boldsymbol{D}}_k)$是模型 $\boldsymbol{W}_k^t$ 在数据集 $\widetilde{\boldsymbol{D}}_k$ 上的本地训练损失,它是通过对数据集 $\widetilde{\boldsymbol{D}}_k$ 中的样本损失进行平均来计算的,即:

$$\ell_k(\boldsymbol{W}_k^t;\widetilde{\boldsymbol{D}}_k)=\frac{1}{|\widetilde{\boldsymbol{D}}_k|}\sum_{(\boldsymbol{x},\boldsymbol{y})\in\widetilde{\boldsymbol{D}}_k}L_{\text{CE}}(\boldsymbol{y},\boldsymbol{p}) \tag{10}$$

其中,$L_{\text{CE}}(\cdot,\cdot)$是样本的交叉熵损失,$\boldsymbol{p}=\phi_{\boldsymbol{W}_k^t}(\boldsymbol{x})$是在模型参数为 $\boldsymbol{W}_k^t$ 时样本 $\boldsymbol{x}$ 的预测结果。

本地训练完成后,客户端将其模型参数 $\boldsymbol{W}_k^{t+1}=\{\boldsymbol{U}_k^{t+1},\boldsymbol{V}_k^{t+1}\}$的特征提取器参数 $\boldsymbol{U}_k^{t+1}$ 上传至服务器。最后,服务器通过加权平均来聚合更新第 $t+1$ 轮的全局特征提取器模型参数。

$$\boldsymbol{U}_{\text{glo}}^{t+1}=\sum_{k\in\boldsymbol{A}^t}\frac{|\widetilde{\boldsymbol{D}}_k|}{\sum\limits_{k=\boldsymbol{A}^t}|\widetilde{\boldsymbol{D}}_k|}\boldsymbol{U}_k^{t+1} \tag{11}$$

重复上述过程,直到设定的训练轮数或模型收敛。此时,各客户端得到全局特征提取器参数,记为 $\boldsymbol{U}_{\text{glo}}$,用于下一阶段的分类器再训练。

BPFed 第一阶段基于 PFL 的表示学习的整体工作流程如算法 1 所示。

算法 1 基于 PFL 的表示学习

输入:全局训练的总轮数 T,选择比例 ρ

输出:训练 T 轮后的全局特征提取器参数 $\mathbf{U}_{\text{glo}}^{\text{T}}$

1. 服务器初始化 $\mathbf{U}_{\text{glo}}^0$
2. for t=0,1,…,T−1 do
3. $\quad\mathbf{A}^t\leftarrow$随机选择$\lceil K\rho\rceil$个客户端
4. $\quad$服务器发送 $\mathbf{U}_{\text{glo}}^t$给 $\mathbf{A}^t$ 中每个客户端
5. $\quad$for each 客户端 $k\in\mathbf{A}^t$ in parallel do

6. 客户端 k 利用式(7)和式(8)进行 Mixup 数据增强
7. 客户端 k 利用式(9)更新本地模型
8. 客户端 k 发送 $\mathbf{U}_{k}^{t+1}$ 给服务器
9. end for
10. 服务器利用式(11)聚合计算 $\mathbf{U}_{glo}^{t+1}$
11. end for
12. 返回 $\mathbf{U}_{glo}^{T}$

在算法 1 中，中央服务器首先初始化全局特征提取器参数 $\boldsymbol{U}_{glo}^{0}$（第 1 行）。在训练过程中，中央服务器连续执行 T 轮迭代。在每一轮迭代 t 中，服务器随机选择一组客户端集合 $\boldsymbol{A}^{t}$，客户端数量为 $\lceil K\rho \rceil$，并将当前的全局特征提取器参数 $\boldsymbol{U}_{glo}^{t}$ 发送给这些客户端（第 3—4 行）。对于集合中的每个客户端 k，其根据式(7)—式(9)应用 Mixup 策略进行数据增强，并通过随机梯度下降法（Stochastic Gradient Descent，SGD）更新本地模型（第 6—7 行）。更新后的本地特征提取器参数 $\boldsymbol{U}_{k}^{t+1}$ 被上传至中央服务器（第 8 行）。每轮迭代结束时，服务器聚合所有客户端的参数更新，依据式(11)更新全局特征提取器参数（第 10 行）。完成 T 轮迭代后，算法返回聚合后的全局特征提取器参数 $\boldsymbol{U}_{glo}^{T}$（第 12 行）。

3.2.2 基于全局特征增强的个性化分类器再训练阶段

1）基于全局协方差矩阵的类级特征增强

在异构长尾情况下，客户端数据可能极度倾斜，头类和尾类的数据量差异巨大，这将导致分类器对尾类泛化能力差而对头类置信过度。因此，在 BPFed 第二阶段，提出采用基于全局协方差矩阵的类级特征增强策略，其核心思想是客户端将本地样本特征的平均协方差矩阵上传到服务器加权聚合，得到全局协方差矩阵，然后客户端以此为扰动构建类平衡的增强特征向量集，作为个性化分类器再训练的输入。具体过程如下：

首先，客户端 $k\in[1,K]$ 通过第一阶段获得的全局特征提取器按类提取本地数据样本的原始特征向量，即：

$$\boldsymbol{z}_{i}^{c}=f_{\mathbf{U}_{glo}}(x_{i}^{c}) \tag{12}$$

其中，$\boldsymbol{x}_{i}^{c}$ 表示客户端 k 的第 c 类数据子集 $\boldsymbol{D}_{k}^{c}$ 中的第 i 个样本，$\boldsymbol{z}_{i}^{c}$ 为样本 $\boldsymbol{x}_{i}^{c}$ 对应的原始特征向量。

则客户端 k 的第 c 类样本特征的协方差矩阵 $\boldsymbol{M}_{k}^{c}\in\mathbb{R}^{d\times d}$ 可表示为：

$$\boldsymbol{M}_{k}^{c}=\frac{1}{n_{k}^{c}-1}\sum_{i=1}^{n_{k}^{c}}(\boldsymbol{z}_{i}^{c}-\bar{\boldsymbol{z}}^{c})(\boldsymbol{z}_{i}^{c}-\bar{\boldsymbol{z}}^{c})^{\mathrm{T}} \tag{13}$$

其中，$\bar{\boldsymbol{z}}^{c}$ 是第 c 类原始特征向量的均值向量，计算式如式(14)所示：

$$\bar{\boldsymbol{z}}^{c}=\frac{1}{n_{k}^{c}}\sum_{i=1}^{n_{k}^{c}}\boldsymbol{z}_{i}^{c} \tag{14}$$

其次，各个客户端 $k\in[1,K]$ 根据每类的协方差矩阵 $\boldsymbol{M}_{k}^{c}$ 计算其本地协方差矩阵 $\boldsymbol{M}_{k}$：

$$\boldsymbol{M}_{k}=\frac{\sum_{c\in C_{k}^{pos}}n_{k}^{c}\boldsymbol{M}_{k}^{c}}{\sum_{c\in C_{k}^{pos}}n_{k}^{c}} \tag{15}$$

然后，每个客户端将这些本地协方差矩阵上传至服务器，并在服务器上进行加权平均聚合以得到全局协方差矩阵 $\boldsymbol{M}_{glo}$：

$$\boldsymbol{M}_{glo}=\frac{\sum_{k=1}^{K}n_{k}\boldsymbol{M}_{k}}{\sum_{k=1}^{K}n_{k}} \tag{16}$$

此全局协方差矩阵随后分发至所有客户端。每个客户端基于多元高斯分布 $N(0,\boldsymbol{M}_{glo})$ 进行采样，以生成相应的扰动。扰动的数量以样本总量最大的类别为准，令 $n_{k}^{max}=\max\{n_{k}^{1},\cdots,n_{k}^{C}\}$ 为客户端 k 拥有样本数最多的类的数量，则客户端 k 第 c 类扰动的数量 $\tilde{n}_{k}^{c}$ 为：

$$\tilde{n}_{k}^{c}=n_{k}^{max}-n_{k}^{c} \tag{17}$$

最后，客户端将这 $\tilde{n}_{k}^{c}$ 个扰动添加到各自类别的原始特征向量中，生成类平衡（每类的特征数量相同）的新特征向量集，并将其用于个性化分类器再训练和失衡校正。

2）基于样本权重的标签感知平滑损失函数

由于交叉熵损失函数倾向于高估模型对头类数据样本的预测置信度，而忽视尾类数据[12]，因此本文提出一种基于样本权重的标签平滑损失函数。该函数采用新的标签平滑策略，能灵活地实现不同的平滑度。同时，该函数只关注本地的拥有类别，有助于模型更好地学习个性化类别特征，并减少计算开销。具体定义如下：

$$L_{smooth}(\boldsymbol{q},\boldsymbol{p})=-\sum_{j\in C_{k}^{pos}}q_{j}\log p_{j} \tag{18}$$

其中，$\boldsymbol{p}$ 是模型的预测概率向量，p_{j} 表示 $\boldsymbol{p}$ 中第 j 个预测概率分量；$\boldsymbol{q}$ 是平滑后的软标签，q_{j} 表示 $\boldsymbol{q}$ 中第 j 个软标签分量。对于任一第 c 类的特征样本，q_{j} 的计算如下：

$$q_{j}=\begin{cases}1-g(n_{k}^{c}), & j=c\\ \dfrac{g(n_{k}^{c})}{|\mathbf{C}_{k}^{pos}|-1}, & j\neq c\end{cases} \tag{19}$$

$$g(n_{k}^{c})=\delta\left(\frac{n_{k}^{c}}{n_{k}^{max}}\right)^{\gamma} \tag{20}$$

其中，$g(\cdot)$ 为新的标签平滑策略函数，能够根据样本数量动态调整各类别的平滑度；δ 与 γ 为一对超参数，用于控制平滑度的大小与样本数量之间的关系。当 $\delta=0$ 时，退化到原本的独热标签。当 $\gamma=1$ 时，$g(\cdot)$ 与样本数量呈线性关系；当 $\gamma<1$ 时，$g(\cdot)$ 呈现凹函数形态；当 $\gamma>1$ 时，$g(\cdot)$ 为凸函数形态。

由此可见，除了 $\delta=0$ 的情况外，对于数据量较大的头类样本，上述策略均会增加其平滑效果，而对于数据量较小的尾类样本，则赋予较低的平滑度。因此，通过最小化损失函数 $L_{smooth}(\boldsymbol{q},\boldsymbol{p})$，能缓解头类过度置信问题并提升尾类泛化能力。

最后，固定全局特征提取器参数，对客户端分类器进行 E 轮个性化再训练。本文采用文献[12]提出的方法，该方法结合了文献[4]的分类器重训练（Classifier Re-training，cRT）和可学习的权重缩放（Learnable Weight Scaling，LWS）两种方法，能够同时在权重参数的方向性和范数上进行优化，提高训练性能。

BPFed 第二阶段基于全局特征增强的个性化分类器再训练的整体工作流程如算法 2 所示。

算法 2 基于全局特征增强的个性化分类器再训练

输入：表示学习阶段得到的全局特征提取器参数 $\mathbf{U}_{glo}$，本地个性化训练的总轮数 E

输出：最终的个性化模型参数 $\{\mathbf{W}_{k}\}$

1. for each k∈[1,K] in parallel do
2. 客户端 k 利用式(15)计算 $\mathbf{M}_k$
3. 客户端 k 发送 $\mathbf{M}_k$给服务器
4. end for
5. 服务器利用式(16)聚合计算 $\mathbf{M}_{glo}$
6. 服务器将 $\mathbf{M}_{glo}$发送给所有客户端
7. for each k∈[1,K] in parallel do
8. 客户端 k 利用式(17)和 $N(0,\mathbf{M}_{glo})$进行扰动采样
9. 在原始特征向量中加入扰动样本获得新的类平衡特征向量集
10. for e= 0,1,…,E do
11. 客户端 k 利用式(18)更新 $\mathbf{V}_k$
12. end for
13. $\mathbf{W}_k \leftarrow \{\mathbf{U}_{glo},\mathbf{V}_k\}$
14. end for
15. 返回$\{\mathbf{W}_k\}$

算法 2 采用第一阶段获得的全局特征提取器参数 $\boldsymbol{U}_{glo}$,每个客户端 k 并行执行以下步骤:首先根据式(15)计算其本地协方差矩阵 $\boldsymbol{M}_k$,并将其上传至中央服务器(第 1—4 行);服务器根据式(16)聚合所有客户端上传的本地协方差矩阵以形成全局协方差矩阵 $\boldsymbol{M}_{glo}$,随后发送给每个客户端(第 5—6 行);每个客户端 k 从多元高斯分布 $N(0,\boldsymbol{M}_{glo})$ 中采样扰动,并根据式(17)将这些扰动添加到原始特征向量中以获得新的类平衡特征向量集(第 7—9 行);接着,在 E 轮本地训练中,客户端 k 根据式(18)更新本地分类器参数 $\boldsymbol{V}_k$,这些参数使用标签平滑损失进行优化(第 10—12 行);训练完成后,每个客户端获得的个性化分类器参数 $\boldsymbol{V}_k$ 与全局特征提取器参数 $\boldsymbol{U}_{glo}$结合,形成最终的个性化模型参数 $\boldsymbol{W}_k$(第 13 行);最后,算法返回每个客户端的个性化模型参数 $\{\boldsymbol{W}_k\}$(第 15 行)。

4 实验

4.1 实验设置

本文在两个图像分类长尾数据集 CIFAR-10-LT 和 CIFAR-100-LT 上评估了 BPFed 算法的性能,并设定不平衡因子(Imbalance Factors,IF)为 100 和 50[23-24]。IF 是最大类别的样本数与最小类别的样本数之比。IF 值越大,不平衡程度越高。接着,采用带有超参数 β 的狄利克雷(Dirichlet)分布来模拟不同程度的数据异构性[25]。β 值越小,异构程度越高。这里设定 β 值为 0.5 和 0.2。

对于图像分类任务,采用 ResNet[26] 网络模型,并将其最后的全连接层定义为分类器,此全连接层之前的所有卷积层及其关联的网络结构被视为特征提取器。在具体实验中,使用 ResNet-8 和 ResNet-18 分别作为 CIFAR-10-LT 和 CIFAR-100-LT 数据集分类的基础模型。

本文基于 PyTorch 实现了 BPFed 框架,所有实验均在一台配置 NVIDIA GeForce RTX 3080 GPU 的服务器上进行。实验模拟一个服务器和一组客户端,设客户端总数 $K=20$,每轮客户端选择比例 $\rho=0.4$。在第一阶段,设共进行 $T=200$ 轮全局通信,批量大小设置为 32,优化器使用学习率为 0.1 的 SGD;在第二阶段,设个性化本地训练的轮数 $E=5$,优化器使用学习率为 0.01,动量为 0.9,权重衰减为 5×10^{-4} 的 SGD。BPFed 的超参数设置如表 1 所列,这些参数值是基于模型训练的最优性能确定的(具体在 4.3.3 节中讨论)。除非另有说明,所有关于 BPFed 的实验都使用表 1 中的超参数设置。

表 1 BPFed 的超参数

Table 1 Hyperparameters of BPFed

Hyperparameters	Value
α	1
δ	0.5
γ	0.4

4.2 对比算法和评估指标

为了评估 BPFed 的性能,选取如下有代表性且相关的 FL 算法进行对比:

1)FedAvg[2]:最基本的基于加权平均的 FL 算法;

2)FedProx[15]:通过引入正则化项来解决 non-IID 问题的经典 FL 算法;

3)FedPer[10]:基于参数解耦的 PFL 算法;

4)BalanceFL[9]:通过在本地训练引入知识继承以及特征增强来解决长尾问题的 FL 算法;

5)CReFF[8]:首个采用两阶段的长尾分类 FL 算法;

6)FedAFA[11]:首个基于对抗性特征增强的长尾分类 PFL 算法。

实验的主要评估指标与 3.1 节的优化目标一致,为所有客户端拥有类的平均测试准确度,即:

$$Accuracy=\frac{1}{K}\sum_{k=1}^{K}acc_k(\phi_{W_k}) \tag{21}$$

其中,$acc_k(\phi_{W_k})$的计算由式(4)给出。

同时,为了评估 PFL 模型的个性化能力,实验还将对每个客户端模型的测试准确度(acc_k)进行分析。

4.3 实验结果

4.3.1 对比实验结果分析

表 2 和表 3 列出了在全局长尾分布且客户端数据异构性场景下,BPFed 与不同算法在 CIFAR-10-LT 和 CIFAR-100-LT 数据集上的准确度比较结果。通过对比分析可以发现,BPFed 在不同长尾不平衡程度(IF)和异构性(β)条件下均表现优异。具体地,由表 2 可见,FedAvg 算法的表现最差,说明其加权平均策略不能有效处理数据的长尾不平衡性和异构性情况。FedProx 通过引入正则化来应对数据异构性问题,其准确度较 FedAvg 有所提升,但不如采用个性化方法的 FedPer。而 FedPer 并未考虑数据的长尾不平衡分布,其性能依然有限。BalanceFL 通过知识继承和特征增强方法缓解不平衡问题,其性能得到进一步提升。与 CReFF 相比,BalanceFL 在异构程度较小($\beta=0.5$)时性能较好,而在异构程度较大($\beta=0.2$)时性能较差。CReFF 则通过两阶段训练策略同时解决数据长尾分布和异构性问题,但其采用的单一全局模型并不能很好地适应客户端个性化条件。FedAFA 首先引入 PFL 方法并通过生成对抗性特征来缓解上述问题的不利影响,其准确度较 FedPer 又有所提升,但其对优化策略的考虑还不够全面。相比之下,本文所提出的 BPFed 在训练的两个阶段都进行了细致的优化策略设计。其在不同IF和β设置

下的平均准确度相较于 FedAvg, FedProx, FedPer, BalanceFL, CReFF 以及 FedAFA, 分别提升了 16.31%, 15.48%, 7.67%, 4.45%, 4.26%, 1.20%。

表 2 BPFed 与对比算法在 CIFAR-10-LT 上的准确度比较

Table 2 Comparing of accuracy of BPFed and comparative algorithms on CIFAR-10-LT

(%)

Algorithm	$IF=50$		$IF=100$	
	$\beta=0.5$	$\beta=0.2$	$\beta=0.5$	$\beta=0.2$
FedAvg	62.96	61.34	55.62	53.06
FedProx	63.01	62.11	56.26	54.92
FedPer	69.73	68.88	68.54	60.38
BalanceFL	75.25	71.26	70.55	63.35
CReFF	73.50	72.62	70.12	64.89
FedAFA	76.48	75.68	73.67	67.56
BPFed	**77.25**	**76.02**	**75.90**	**69.03**

表 3 BPFed 与对比算法在 CIFAR-100-LT 上的准确度比较

Table 3 Comparing of accuracy of BPFed and comparative algorithms on CIFAR-100-LT

(%)

Algorithm	$IF=50$		$IF=100$	
	$\beta=0.5$	$\beta=0.2$	$\beta=0.5$	$\beta=0.2$
FedAvg	36.27	34.68	31.41	30.09
FedProx	38.01	35.98	32.45	31.84
FedPer	43.39	42.83	40.06	38.05
BalanceFL	44.84	43.33	40.18	38.24
CReFF	40.95	37.83	36.67	34.29
FedAFA	50.97	47.24	45.23	43.12
BPFed	**51.49**	**50.85**	**47.97**	**46.38**

表 3 的结果与表 2 类似, BPFed 在不同设置下的平均准确度相较于上述对比算法分别提升了 16.06%, 14.60%, 8.09%, 7.53%, 11.74%, 2.53%。不同的是, 表 3 中 CReFF 的表现反而不如 FedPer 和 BalanceFL, 这说明 CReFF 所采用的策略并不适合处理类别数量较大的不平衡数据(如 CIFAR-100-LT 数据集)。

为评估模型的个性化能力, 图 2 给出了在 CIFAR-10-LT 数据集上, $IF=50$, $\beta=0.5$ 时, 不同(包括 FedProx, FedPer, CReFF, FedAFA 和 BPFed)模型在各客户端上的准确度。可以看到, 由于长尾异构数据分布的不平衡性, 在不同客户端上, 不同模型的准确度均出现了一定的波动。但是, BPFed 在大多数客户端上的准确度都高于其他算法, 这进一步验证了其在处理长尾异构问题上的有效性。

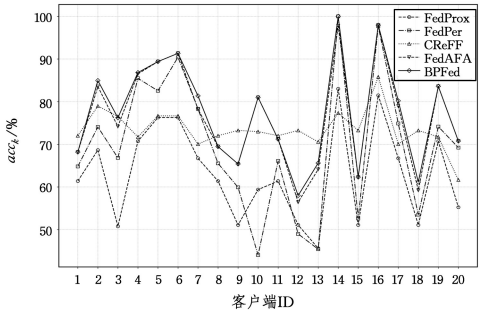


图 2 不同算法模型在各客户端上的准确度

Fig. 2 Accuracy of different algorithmic models on each client

4.3.2 消融实验结果分析

本小节通过消融实验评估了 BPFed 中各策略组件的有效性。实验均在 $IF=50$, $\beta=0.5$ 的 CIFAR-10-LT 数据集上进行。表 4 列出了 BPFed 消融实验的结果, 其中, Mixup (stg. 1)代表第一阶段采用的 Mixup 数据增强方法, FA(stg. 2)代表第二阶段提出的基于全局协方差矩阵的类级特征增强策略, ☑和☒分别表示是否采用相应的策略, Loss(stg. 2)代表第二阶段个性化分类器再训练的损失函数, CE 表示交叉熵损失函数, LS 表示所提出的基于样本权重的标签平滑损失函数, "All classes"表示对所有类进行 LS 计算, "Owned classes"表示仅对客户端本地拥有类进行 LS 计算。

表 4 BPFed 消融实验结果

Table 4 Ablation results of BPFed

Mixup (stg. 1)	FA (stg. 2)	Loss (stg. 2)	Accuracy/%
☑	☒	CE	71.09
☑	☒	CE	72.86
☑	☒	CE	74.11
☑	☒	LS(All classes)	76.13
☑	☒	LS(Owned classes)	**77.25**

由表 4 可知, 基础模型(即不使用 Mixup 和 FA、第二阶段采用 CE 损失函数)的准确率为 71.09%, 这表明采用简单的"先联邦训练后客户端模型微调"两阶段方法虽然有一定效果, 但其准确率仍是最低的, 还不足以应对长尾异构 FL 环境下客户端数据的极度不平衡情况。相比于基础模型, 仅使用 Mixup 的准确率提升了 1.77%, 这验证了 Mixup 有助于提高长尾分布数据特征表示学习模型的泛化性能。进一步地, 同时采用 Mixup 和 FA 能将准确率提高到 74.11%, 这表明在第二阶段使用 FA 得到的类平衡特征空间能有效缓解客户端数据的长尾异构倾斜, 校正失衡的分类器决策边界, 从而进一步提升模型性能。而使用 LS 损失函数相较于使用 CE 的准确率更高, 这表明标签平滑能有效纠正头类置信过度及尾类泛化能力过低的问题, 特别是使用 LS(Owned classes)损失函数时, 准确率达到最高的 77.25%, 与使用 LS(All classes)相比提升了 1.12%。这说明标签平滑在对拥有类别进行优化时, 效果更为显著。

因此, BPFed 框架所采用的 Mixup 和 FA 增强策略以及 LS(Owned classes)标签平滑损失函数能有效解决长尾异构数据的 FL 问题, 显著提高模型性能。

4.3.3 超参数的影响分析

在 BPFed 中存在 3 个超参数: 用于 Mixup 策略中的混合超参数 λ 取值的 Beta 分布参数 α、标签平滑损失函数里控制平滑度的两个超参数 δ 与 γ。本小节同样在 $IF=50$, $\beta=0.5$ 的 CIFAR-10-LT 数据集上进行多组实验, 来评估它们对模型性能的影响。

图 3 展示了 α 取不同值时 BPFed 的性能。由图可见, 当 $\alpha=1$ 时, 即 λ 服从(0,1)均匀分布时, BPFed 能获得最高的准确度, 比 $\alpha=0$(相当于不采用 Mixup 策略)时的模型性能提升了 1.40%。而当 $\alpha>1$ 时, 输入样本被平均混合的比例将随之增加, 这在一定程度上降低了样本的多样性。

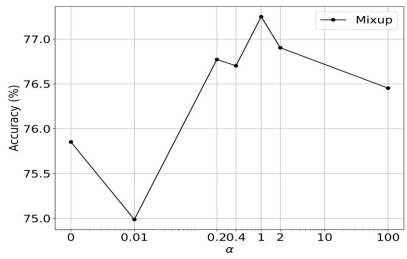


图 3 Mixup 超参数对 BPFed 的影响

Fig. 3 Impact of Mixup hyperparameters on BPFed

图 4 给出了 δ 与 γ 取不同值时 BPFed 的性能矩阵。在多分类系统中，若该样本某一类别的预测概率超过 0.5，则分类器将该样本归类为这一类。为确保超参数的合理性，本文设置 $\delta \in [0, 0.5]$。实验结果显示，当标签平滑策略函数选择 $\gamma=0.4$，$\delta=0.5$（图 4 中的白色方框）时，能获得最佳的标签平滑度，并使 BPFed 模型达到最高的准确度 77.25%。

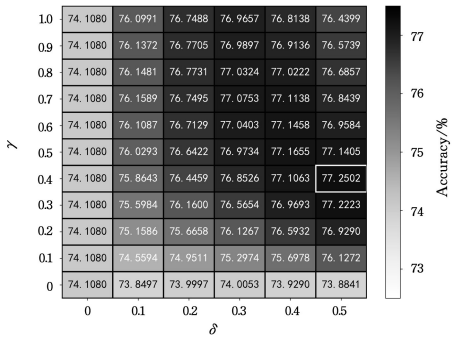


图 4 标签平滑超参数对 BPFed 的影响

Fig. 4 Impact of label-aware smoothing hyperparameters on BPFed

结束语 本文提出了一种面向长尾异构数据的 BPFed 框架，并在各训练阶段综合运用 Mixup 数据增强、参数解耦的 PFL 训练、类级特征增强、标签平滑损失函数等多种策略，进一步优化其性能。实验结果表明，在不同长尾和异构条件下，BPFed 与其他代表性对比算法相比，具有显著的优越性。此外，消融和超参数影响实验进一步验证了所提策略和方法的有效性。

当前 BPFed 框架中所使用的数据/特征增强方法还很有限，下一步，可研究采用更广泛的（如生成对抗性网络等）方法，以提高数据/特征增强的有效性。同时，可尝试定义不同形式的损失函数，以验证其解决长尾异构问题的能力。另外，BPFed 主要关注单一领域内的长尾异构数据处理，在未来的工作中，可探索将 BPFed 应用于跨领域或跨数据源的场景，研究不同领域数据的有效融合策略，以及如何通过跨领域 FL 提升模型的泛化能力。

参 考 文 献

[1] MCENROE P, WANG S, LIYANAGE M. A survey on the convergence of edge computing and AI for UAVs: Opportunities and challenges [J]. IEEE Internet of Things Journal, 2022, 9(17): 15435-15459.

[2] MCMAHAN B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data [C]// Proceedings of the20th International Conference on Artificial Intelligence and Statistics(AISTATS). New York: PMLR, 2017: 1273-1282.

[3] ZHANG Y, KANG B, HOOI B, et al. Deep long-tailed learning: A survey [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(9): 10795-10816.

[4] KANG B, XIE S, ROHRBACH M, et al. Decoupling Representation and Classifier for Long-Tailed Recognition [C]// Proceedings of International Conference on Learning Representations. Washington DC: ICLR, 2020: 1-16.

[5] LIN T, GOYAL P, GIRSHICK R, et al. Focalloss for dense object detection [C]// Proceedings of International Conference on Computer Vision (ICCV). Piscataway, NJ: IEEE, 2017: 2980-2988.

[6] ZHANG H, CISSE M, DAUPHIN Y N, et al. Mixup: Beyond empirical risk minimization [C]// Proceedings of International Conference on Learning Representations. Washington DC: ICLR, 2018: 1-13.

[7] YIN X, YU X, SOHN K, et al. Featuretransfer learning for face recognition with under-represented data [C]// Proceedings of International Conference on Computer Vision and Pattern Recognition(CVPR). Piscataway, NJ: IEEE, 2019: 5704-5713.

[8] SHANG X, LU Y, HUANG G, et al. Federatedlearning on heterogeneous and long-tailed data via classifier re-training with federated features [C]// Proceedings of International Joint Conference on Artificial Intelligence. Freiburg: IJCAI, 2022: 2218-2224.

[9] SHUAI X, SHEN Y, JIANG S, et al. BalanceFL: Addressing class imbalance in long-tail federated learning [C]// Proceedings of the 21st ACM/IEEE International Conference on Information Processing in Sensor Networks(IPSN). Piscataway, NJ: IEEE, 2022: 271-284.

[10] ARIVAZHAGAN M G, AGGARWAL V, SINGH A K, et al. Federated lear-ning with personalization layers [J]. arXiv: 1912.00818, 2019.

[11] LU Y, QIAN P, HUANG G, et al. Personalized federated learning on long-tailed data via adversarial feature augmentation [C]// Proceedings of 2023 IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP). Piscataway, NJ: IEEE, 2023: 1-5.

[12] ZHONG Z, CUI J, LIU S, et al. Improving Calibration for Long-Tailed Recognition [C]// Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2021: 16489-16498.

[13] WANG X M, FANG R. Device selection in federated learning under class imbalance [J]. Application Research of Computers, 2021, 38(10): 2968-2973.

[14] WU F, SONG Y B, JI Y M, et al. Contribution-based federated learning approach for global imbalanced problem [J]. Computer Science, 2023, 50(12): 343-348.

[15] LI T, SAHU A K, ZAHEER M, et al. Federated Optimization in Heterogeneous Networks [C]// Proceedings of International Conference on Machine Learning Systems (ICML). New York: IMLS, 2020: 429-450.

[16] KARIMIREDDY S P, KALE S, MOHRI M, et al. Scaffold: Stochastic controlled averaging for federated learning [C]// Proceedings of International Conference on Machine Learning (ICML). New York: IMLS, 2020: 5132-5143.

[17] LI Q, HE B, SONG D, et al. Model-Contrastive Federated Learning [C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2021: 10713-10722.

[18] ACAR D A E, ZHAO Y, NAVARRO R M, et al. Federated Learning Based on Dynamic Regularization [C]// Proceedings of International Conference on Learning Representations. Washington DC: ICLR, 2021: 1-14.

[19] CHAI Z, ALI A, ZAWAD S, et al. Tifl: A tier-based federated learning system [C]// Proceedings of the 29th International Symposium on High-Performance Parallel and Distributed Computing (HPDC). New York: ACM, 2020: 125-136.

[20] ZHAO Y, LI M, LAI L, et al. Federated learning with non-IID data [J]. arXiv: 1806.00582, 2018.

[21] LUO M, CHEN F, HU D, et al. No fear of heterogeneity: Classifier calibration for federated learning with non-IID data [J]. Advances in Neural Information Processing Systems, 2021, 34: 5972-5984.

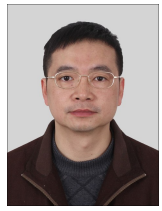
[22] LIU Y, WU X, LIU C K. Research onpersonalized federated learning algorithm in industrial Internet of things [J]. Journal of Chinese Computer Systems, 2025, 46(1): 209-216.

[23] KRIZHEVSKY A. Learning multiple layers of features from tiny images [R]. University of Toronto, 2009.

[24] CAO K, WEI C, GAIDON A, et al. Learningimbalanced datasets with label-distribution-aware margin loss [C]// Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS). New York: PMLR, 2019: 1-12.

[25] HSU T M H, QI H, BROWN M. Measuring the effects of non-identical data distribution for federated visual classification [J]. arXiv: 1909.06335, 2019.

[26] HE K, ZHANG X, REN S, et al. Deepresidual learning for image recognition [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ: IEEE, 2016: 770-778.



WU Jiagao, born in 1969, Ph.D, associate professor, is a member of CCF (No. 12107M). His main research interests include computer network, edge computing and artificial intelligence.

(责任编辑: 何杨)