计 算 机 研 究 与 发 展
Journal of Computer Research and Development

https://crad.ict.ac.cn
63(4): 998−1009, 2026

FedDW: 基于一致性优化的权重蒸馏异构联邦学习方法

刘佳瑜[1] 王 勇[1,2] 杨 静[1] 王念滨[1,2]

[1](哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)

[2](电子政务建模仿真国家工程实验室(哈尔滨工程大学) 哈尔滨 150001)

(jiayuliu@hrbeu.edu.cn)

FedDW: Distilling Weights Through Consistency Optimization in Heterogeneous Federated Learning

Liu Jiayu[1], Wang Yong[1,2], Yang Jing[1], and Wang Nianbin[1,2]

[1](*College of Computer Science and Technology*, *Harbin Engineering University*, *Harbin* 150001)

[2](*Modeling and Emulation in E-Government National Engineering Laboratory* (*Harbin Engineering University*), *Harbin* 150001)

Abstract Federated learning is a classic distributed machine learning paradigm that enables collaborative model training without centralizing data. This approach offers significant advantages in ensuring data privacy. However, it faces numerous challenges in terms of training efficiency and model performance due to the significant data heterogeneity across clients and the increasing scale of federated networks. Existing studies have shown that, in independent and identically distributed (IID) environments, the model parameter structure typically adheres to specific consistency relations, which are often preserved in the intermediate results during the neural network training process. If these consistency relations can be identified and regularized under non-independent and identically distributed (non-IID) data, it can help align the parameter distribution with the IID scenario, thereby mitigating the impact of data heterogeneity. Based on this idea, we introduce the concept of deep learning on encrypted data and propose a consistency optimization paradigm. We further explore the consistency relationship between data soft labels and the model's classification layer weight matrix, from which we construct a new heterogeneous federated learning framework, FedDW. Experiments are conducted on four public datasets and several neural network models, including ResNet and ViT. The results show that, under highly heterogeneous client data settings, FedDW outperforms 10 state-of-the-art federated learning methods with an average accuracy improvement of approximately 3%. Moreover, we theoretically prove that FedDW offers higher training efficiency, with the additional backpropagation computation overhead being negligible.

Key words federated learning; data heterogeneity; deep learning encrypted data; consistency optimization paradigm; weight distillation

摘 要 联邦学习是一种经典的分布式机器学习范式,允许在不集中数据的前提实现模型协同训练。该方法在保障数据隐私方面具有显著优势,然而由于客户端间数据异质性显著以及联邦规模不断扩大,对

收稿日期: 2025-06-06; **修回日期**: 2025-10-21

基金项目: 国家重点研发计划项目(2022YFC3301804); 教育部人文社会科学研究青年基金项目(20YJCZH172); 黑龙江省自然科学基金项目(LH2024F035); 黑龙江省省属本科高校优秀青年教师基础研究项目(YQJH2023302)

This work was supported by the National Key Research and Development Program of China (2022YFC3301804), the Youth Fund Project of Humanities and Social Sciences Research of the Ministry of Education of China (20YJCZH172), the Heilongjiang Provincial Natural Science Foundation of China (LH2024F035), and the Basic Research Support Program for Outstanding Young Teachers in Provincial Undergraduate Universities in Heilongjiang Province(YQJH2023302).

通信作者: 王勇(wangyongcs@hrbeu.edu.cn)

训练效率和模型性能提出诸多挑战。已有研究表明，在独立同分布（independent and identically distributed, IID）环境下，模型的参数结构通常满足特定一致性关系，这些一致性关系往往在神经网络训练过程的中间结果中得以保留。若在非独立同分布（non-independent and identically distributed, non-IID）数据下能够识别并正则化上述一致性关系，有助于将参数分布向IID情形对齐，从而缓解数据异质性带来的影响。基于上述思想，引入深度学习加密数据概念，并基于此提出了一种一致性优化范式。发掘数据软标签与模型分类层权重矩阵之间的一致性关系，据此构建了一个新的异构联邦学习框架FedDW。在4个公开数据集及多种神经网络模型（包括ResNet, ViT）上开展实验。结果表明，在高度异构的客户端数据设置下，FedDW相较于10种主流联邦学习方法在平均准确率上提升了约3个百分点。此外从理论上证明了FedDW具备更高的训练效率，其附加的反向传播计算开销可以忽略不计。

关键词 联邦学习；数据异质；深度学习加密数据；一致性优化范式；权重蒸馏

中图法分类号 TP183；TP309

DOI: 10.7544/issn1000-1239.202550451 **CSTR:** 32373.14.issn1000-1239.202550451

联邦学习（federated learning, FL）[1-2]是一种注重隐私保护的机器学习范式，能够使多个客户端在不共享原始数据的前提下协同训练模型，近年来引起了广泛关注。传统的联邦学习框架（如FedAvg[3]）通常假设客户端之间的数据分布相似及计算能力均衡。然而，在实际应用中，客户端间的数据异质性与算力差异普遍存在，严重制约了联邦学习系统的效率与模型性能。

数据异质性[4-5]，又称为非独立同分布（non-independent and identically distributed, non-IID），会导致联邦模型性能下降，并增加不同客户端本地更新之间的差异性[6]。类似于长尾数据问题[7-8]，当某类样本在数据集中占比过高时，模型容易偏向该类数据，从而削弱对整体数据分布的建模能力。当前异构联邦学习的主要应对策略是实现客户端之间的信息共享，以便训练出具有全局泛化能力的模型。然而，为实现该目的，客户端需了解其他客户端的数据信息，这往往违背了隐私保护原则。为解决这一矛盾，本文引入了深度学习加密（deep learning encrypted, DLE）数据，DLE数据是在神经网络训练过程中生成的，用于封装训练数据的特征信息。但在当前技术条件下，该数据无法被有效解码以重构原始样本，亦无法推导出训练集的具体统计结果。目前常见的DLE数据主要包括以下3类。1）特征向量[9-11]。利用神经网络模型提取器将高维本地数据映射为低维向量，实现了不可逆压缩。既保留核心特征又降低了共享成本。是目前DLE数据的主流形式。2）软标签[12-14]。该类数据利用知识蒸馏方法进行处理，更注重不同类别之间的相对关系，而非原始样本本身。3）神经网络模型参数[15]。神经网络在特定数据集上训练后的参数可视作封装了其数据特性。共享参数作为一种“黑箱”式的数据表征，借助其不可解释性与不可逆性，实现了一种“以深度学习缺陷变优势”的加密方式。

为此本文提出了一种基于DLE数据的一致性优化方法，具体定义如下。在独立同分布（IID）数据分布下，寻找2类具有一致性关系的DLE数据e_1与e_2。对第n个客户端而言，其一致性目标优化为$|\phi(e_1^{\text{global}})-\psi(e_2^n)|$，其中$e_1^{\text{global}}=Agg(e_1^1,e_1^2,\cdots,e_1^N)$，$\phi$与$\psi$表示映射函数，$Agg(\cdot)$为服务器执行的聚合操作。整个DLE一致性优化过程如图1所示。

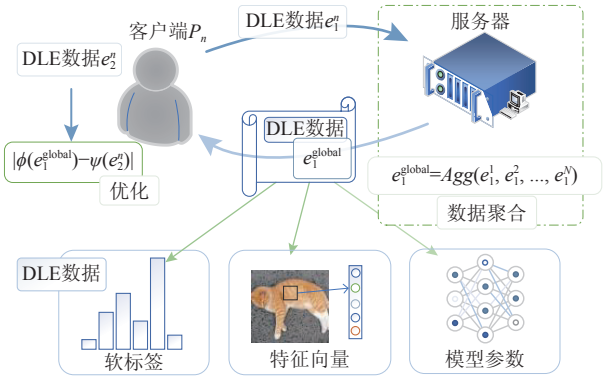


Fig. 1 Using DLE data for regular optimization

图1 使用DLE数据进行正则优化

客户端使用DLE数据进行一致优化时生成e_1和e_2两种类型的DLE数据。e_1用于全局聚合，以捕获全局信息；e_2通过与e_1对齐，引导优化实现全局泛化。e_1必须经过精心选择，以确保其在聚合后仍能保持泛化能力。具体而言，e_1和e_2可以相等。本文识别出软标签与分类层参数之间的一致性关系，并据此提出了一种新的联邦学习框架FedDW。FedDW利用

全局软标签对本地模型的分类层参数进行正则化，在 non-IID 分布下保持模型参数更接近 IID 环境中的分布，从而有效提升模型的泛化能力。FedDW 并未直接使用最常见特征向量作为 DLE 数据，而是观察到模型分类层参数矩阵中每一个类别所对应的判别向量和特征向量表达相似的关系，从而将其替代使用。这样可以节省内存，同时可以减小计算开销。

1 相关工作

在研究发现梯度传输同样可能泄露本地信息之后[16]，FedAvg 被广泛采用为联邦学习中最有效的策略之一。该方法实现简单、性能优越。然而在面对异构本地数据时，FedAvg 表现出明显的局限性。随着数据异质性的加剧，其性能将迅速下降。为应对这一问题，FedProto[11] 引入了服务器聚合的平均特征向量，用于对客户端本地训练进行正则化；FedGH[9] 将这些特征向量直接作为训练数据，用于训练模型的分类器部分。FedTGP[10] 则融合了对比学习的思想[17-19]，通过共享可训练的全局原型和自适应边界对比学习机制，提升来自不同客户端的异构数据与模型的联邦性能。FedKTL[20] 设计了一种机制，通过服务器生成的图像向量对实现对预训练知识向客户端的有效迁移。

此外，FedProx[21]，MOON[22]，FedDyn[23] 是典型的主要以模型参数为共享信息的联邦学习框架。它们在本地目标函数中引入近端项以约束本地更新不偏离初始的全局模型。FedHEAL[24] 则在本地训练过程中维持本地模型与全局模型之间的方差距离，从而提升模型稳定性。

在主要以软标签作为共享信息的代表性联邦学习方法包括 PerAda[12]，MH-pFLID[13]，Fedistill[14]，FedAUX[25]。这类方法普遍通过聚合所有客户端生成的平均软标签，并将其以不同策略应用于本地模型训练来提升模型对异构数据的适应性。

还有一些方法未使用任何 DLE 数据，却依然展现出一定程度的异构性缓解能力，例如 FedRep[26]，FedALA[27]，FedPer[28]，FedAMN[29]。这些方法通常归类于个性化联邦学习(personalized federated learning，pFL)[30]。个性化联邦学习的特点在于不生成统一模型，而是根据客户端数据分布提供个性化模型。然而，深度学习模型的训练往往需要大量数据，而个性化联邦学习主要依赖于本地有限的数据进行训练或微调，这限制了其泛化能力。因此，个性化联邦学习无法在服务器端提供统一的全局模型，也在一定程度上限制了其实际应用范围。

虽然使用 DLE 数据能够有效缓解数据异质性问题，但也不可避免地引入了额外的计算开销。目前已有的方法普遍未对这一问题做出优化。随着模型规模的日益庞大，特别是在大模型兴起的背景下[31-32]处理异构性的附加计算成本已不可忽视。本文提出的方法中用于处理异构问题的正则项只涉及分类层参数。在其训练过程中反向传播的梯度只在最后一层传播，并不流向其他层，从而保证模型大小变化与正则项开销无关。

2 异构联邦学习中基于一致性优化的权重蒸馏方法

2.1 问题陈述

假设系统中共有N个客户端，分别记$P_1,P_2,\cdots,P_N$。每个客户端P_n拥有本地数据集D^n。目标是在不交换原始数据的前提下，借助中心服务器在联合数据集D上训练一个深度神经网络模型θ，优化目标为

$$\arg\min_{\theta}\mathcal{L}(\theta)=\sum_{i=1}^{N}\frac{|D^n|}{|D|}\mathcal{L}^n(\theta)\,, \tag{1}$$

其中$\mathcal{L}^n(\theta)$表示客户端P_n的本地模型损失函数。为了分析方便，本文借鉴文献 [33] 的思想，将深度神经网络模型划分为特征提取器f_ϕ和分类器f_c。进一步地，将分类器f_c拆分为映射层f_π和分类层f_ω，其中分类层f_ω是分类器的最后一层网络参数，其余部分归为映射层f_π。因此，整个模型可以表示为$f_\theta=f_\phi\circ f_\pi\circ f_\omega$，其中$\circ$表示该函数复合(复合的顺序是从左到右)。

2.2 分布匹配

FedDW 的核心思想受到对比学习的启发。在对比学习中，数据x被映射为一个特征向量$\boldsymbol{h}\in\mathbb{R}^k$，数据样本$x_i$和$x_j$之间的相似性可以通过其对应的特征的余弦相似度来表示

$$Sim(\boldsymbol{h}_i,\boldsymbol{h}_j)=\frac{\boldsymbol{h}_i^{\mathrm{T}}\boldsymbol{h}_j}{\|\boldsymbol{h}_i\|\|\boldsymbol{h}_j\|}\,。$$

当移除分类层中的偏置项后，若使用特征向量$\boldsymbol{h}$进行分类层计算，其公式为

$$\boldsymbol{\omega h}=\left(\boldsymbol{\omega}_{c_1},\boldsymbol{\omega}_{c_2},\cdots,\boldsymbol{\omega}_{c_{|C|}}\right)^{\mathrm{T}}\boldsymbol{h}\,, \tag{2}$$

其中c_i表示第i个类别，C表示所有类别。

权重矩阵$\boldsymbol{\omega}\in\mathbb{R}^{|C|\times k}$的第$i$行表示类别$c_i$的判别向量。从对比学习角度来看，$\boldsymbol{\omega}_{c_i}$应属于$c_i$类的特征空间，并通过与$\boldsymbol{h}$的内积计算类别概率。在 IID 分布下，

可以省略模长的归一化操作，内积可以近似为相似性。因为经验观察表明，神经网络分类层中权重向量的模长$|\boldsymbol{\omega}_{c_i}|$与类别$c_i$的数据比例相关。本文设计了实验来验证这一点。使用 MNIST，FashionMNIST，CIFAR-10 数据集，在去除模型偏置项的情况下改变数据集的类别比例进行模型训练。具体而言，针对每个数据集分别进行了 100 次独立实验，每一次实验所使用的类别分布比例均不相同，以充分覆盖不同的类别不均衡情况。考虑到篇幅限制，本文从每个数据集的实验结果中随机选取了 3 个不同类别分布（K_1，K_2，K_3）进行展示。实验结果如图 2 所示。实验结果发现分类层的权重模长与其对应类别在数据集的比例呈负相关。即数据占比越大，模长越小。在 3 个数据集上进行的 100 次实验中，仅观察到 3 个反例（下划线数字显示反例，图中只列出了 1 个反例）。这一反直觉现象说明数据集类别分布会影响分类层的权重向量模长。那么可以定义特征空间中的 2 个性质：模长（反映数据集类别比例）与内积（反映各类别间相似性）。当$\boldsymbol{\omega}_{c_i}$输入到分类层时，输出$c_i$的概率最大，其余为类别间相似度。可以构造 CR（class relation）矩阵$\boldsymbol{\omega}\boldsymbol{\omega}^{\mathrm{T}}$表示类别$c_i$和$c_j$之间的相似性。在 IID 设置下，CR 矩阵与软标签一致；而在 non-IID 情况下，使用全局软标签对 CR 矩阵进行正则化来缓解数据异构性。

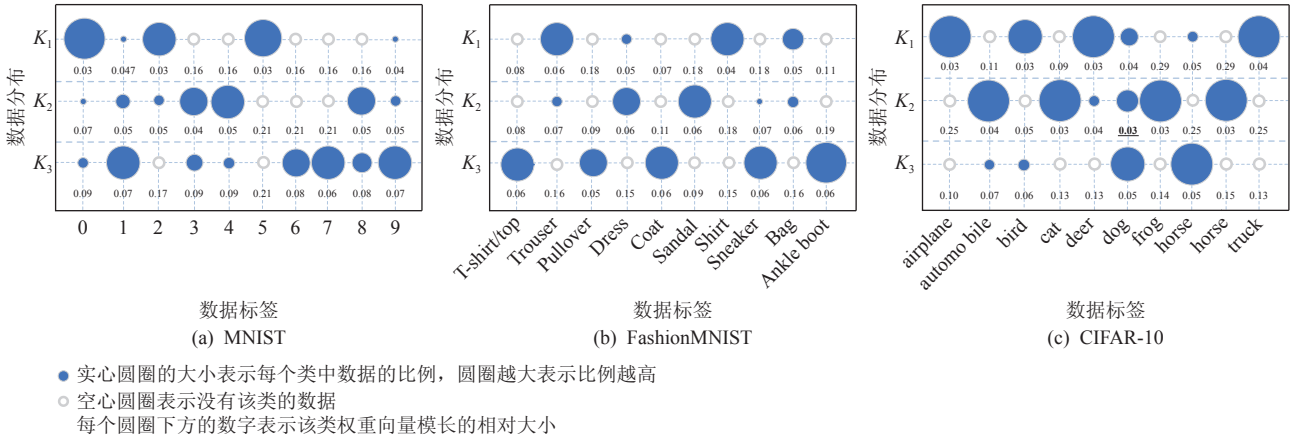


Fig. 2 The relationship between module length and data volume

图 2 模长与数据量之间的关系

可以注意到 FedDW 实际上是将模型分类层的判别向量当作了每一个类别所对应的特征向量来使用。这使得 FedDW 只需要额外存储一种 DLE 数据就可以同时对 2 种 DLE 数据进行正则，在计算效率以及内存占用方面，都要比直接使用特征向量更有优势。

最后，本文通过可视化实验验证了假设：$\boldsymbol{\omega}_{c_i}$属于特征空间中$c_i$类数据的聚类。在 MNIST 数据集上训练一个包含 2 个卷积层和 3 个线性层的神经网络后，使用 t-SNE 对每个类的特征向量和权重向量进行可视化。如图 3 所示，图中使用“×”表示分类层中每类数据的权重向量，“×”的交叉点表示权重向量在可视化空间中的具体位置。可以看到，每个权重向量都属于相应类的簇。分类层权重与各自的类聚类对齐，验证了上述假设。

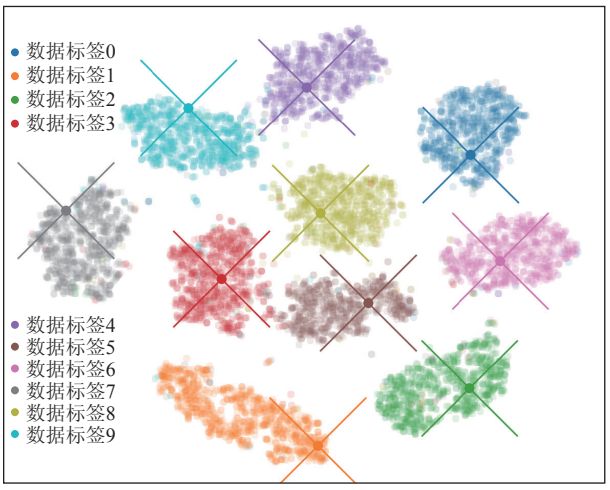


Fig. 3 Visual experiment

图 3 可视化实验

2.3 FedDW 框架

1）全局聚合。在第t轮通信中，服务器按类标签对客户端上传的软标签进行加权平均，得到全局平均软标签矩阵$\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}} \in \mathbb{R}^{|C|\times|C|}$（SL 矩阵）。为降低通信带宽，每个客户端使用相同方式构建本地平均 SL 矩阵，并将其与本地数据集中各类样本数量一起发送给服务器。客户端P_n的本地平均 SL 矩阵$\bar{\boldsymbol{\Omega}}_t^n \in \mathbb{R}^{|C|\times|C|}$计算如下：

$$\bar{\Omega}_t^n[i,j]=\frac{1}{|D_i^n|}\sum_{(x,y)\in D_i^n\wedge y=c_i}Q_{t,j}^n, \tag{3}$$

$$\boldsymbol{Q}_t^n=\sigma\left(f_{\phi_t^n}\circ f_{\tau_t^n}\circ f_{\omega_t^n}(x)\right), \tag{4}$$

$$\boldsymbol{Q}_t^n=\left(Q_{t,1}^n,Q_{t,2}^n,\cdots,Q_{t,|C|}^n\right)^{\mathrm{T}}, \tag{5}$$

其中σ是 Softmax 操作。

注意上传平均 SL 矩阵可能存在用户隐私泄露的风险。可通过多种方法解决，例如服务器提供公开数据集[12]或使用差分隐私技术[34]。本文对此不作详细讨论。服务器根据客户端上传的信息计算全局 SL 矩阵的方式为：

$$\bar{\Omega}_{t+1}^{\mathrm{global}}[i,j]=\sum_{n=1}^{N}\frac{|D_i^n|}{|D_i|}\bar{\Omega}_t^n[i,j], \tag{6}$$

其中D_i^n表示客户端P_n中第i类样本的数量。除了获取全局 SL 矩阵，服务器还会像 FedAvg 一样对模型参数执行加权平均：

$$\theta_{t+1}^{\mathrm{global}}=\sum_{n=1}^{N}\frac{|D^n|}{|D|}\theta_t^n。 \tag{7}$$

2)客户端本地训练。在联邦学习训练过程中，每个客户端使用自己的数据并行独立训练本地网络参数。在第t轮通信中，客户端P_n从服务器接收全局模型$\theta_t^{\mathrm{global}}$和全局 SL 矩阵$\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}}$，用全局模型替换其本地模型进行本地训练。全局 SL 矩阵用于训练期间的正则化。局部训练损失函数分为 2 部分。第 1 个是分类损失$\mathcal{L}_{\mathrm{cla},t}^n$。它是用局部数据集训练神经网络参数的损失。对于分类问题损失计算为

$$\mathcal{L}_{\mathrm{cla},t}^n=E_{(x,y)\sim D^n}[\ell(\theta_t^n;(x,y))], \tag{8}$$

$$\ell(\theta_t^n;(x,y))=-\sum_{i=1}^{|C|}\log(Q_{t,i}^n)[y==c_i]。 \tag{9}$$

第 2 个是正则化损失$\mathcal{L}_{\mathrm{reg}}^n$，它使用全局 SL 矩阵来正则训练。根据上述分析，可知在去除最后一个线性层的偏差项后，在 IID 数据分布下，CR 矩阵和全局 SL 矩阵具有相同的数值意义，此属性用于正则 non-IID 分布下的参数。那么在 DLE 一致性优化中，第 1 种 DLE 数据是 SL 矩阵，第 2 种 DLE 数据是 CR 矩阵，CR 矩阵基于 SL 矩阵进行正则化，然后进行 Softmax.Frobenius 距离测量矩阵之间的差异，并将其作为正则化项。这确保了最终的神经网络参数保持 IID 数据的属性，调整其他参数以减轻数据异构性。正则化计算公式为

$$\mathcal{L}_{\mathrm{reg},t}^n(\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}},\boldsymbol{\omega}_t^n)=\frac{1}{|C|^2}\left\|\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}}-\sigma(\boldsymbol{\omega}_t^n(\boldsymbol{\omega}_t^n)^{\mathrm{T}})\right\|_{\mathrm{F}}^2, \tag{10}$$

这里σ是对矩阵行的 Softmax 运算。

综合上述 2 个损失项，在第t次通信轮中，客户端本地模型更新的目标函数为

$$\mathcal{L}_t^n:=\mathcal{L}_{\mathrm{cla},t}^n\left(D^n;\theta_t^n\right)+\mu\mathcal{L}_{\mathrm{reg},t}^n\left(\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}};\boldsymbol{\omega}_t^n\right), \tag{11}$$

其中μ是调节分类损失与正则项之间权衡的超参数。本地模型的参数更新按照以下梯度下降公式进行：

$$\theta_t^n\leftarrow\theta_t^n-\eta\nabla_{\theta_{t+1}^n}\mathcal{L}_t^n\left(D^n;\theta_t^n;\bar{\boldsymbol{\Omega}}_t^{\mathrm{global}}\right), \tag{12}$$

其中η是学习率。整个方法过程如图 4 所示。

3 理论分析

3.1 收敛性分析

为了分析 FedDW 的理论性质，本文引入有序对$\langle t,r\rangle$，表示第t轮中的第r次迭代。其中t表示通信轮次，$r\in\left\{\frac{1}{2},1,\cdots,R\right\}$表示本地更新的迭代次数。而

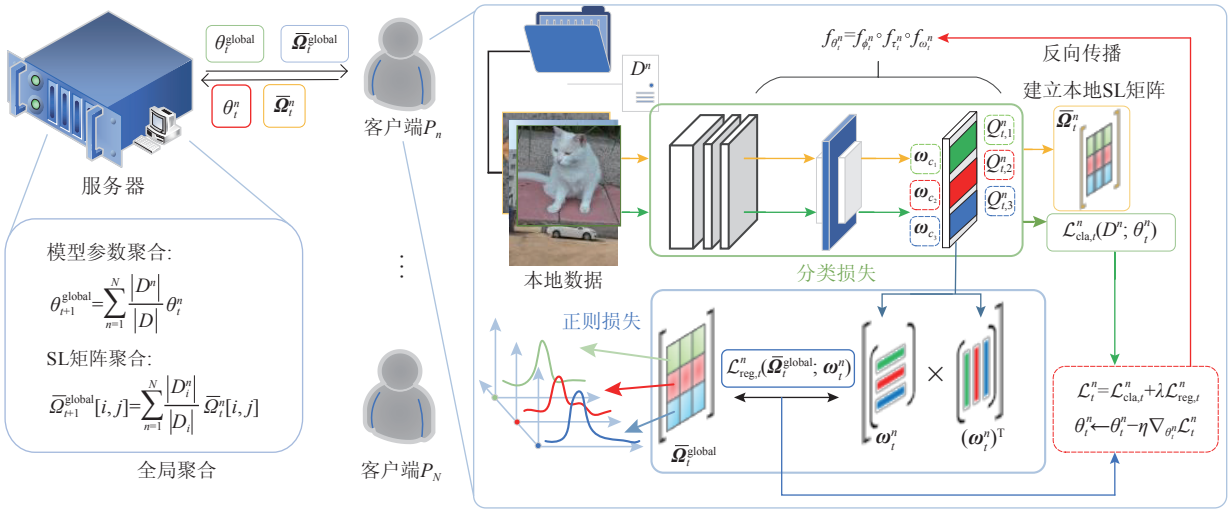


Fig. 4 The entire training process of FedDW

图 4 FedDW 的整个训练过程

$\left\langle t,\frac{1}{2}\right\rangle$表示在第$t$轮开始时，客户端已完成将自身模型替换为全局模型，并成功接收了$\tilde{\boldsymbol{\Omega}}_t^{\text{global}}$。沿用以往的收敛性分析方法[9,11,21]，下面提出以下3个关于模型参数梯度的统计学假设。

假设 1. Lipschitz平滑性。第n个本地损失函数是L_1-Lipschitz平滑的，因此其梯度是L_1-Lipschitz连续的。其中$L_1>0$，对所有$r_1,r_2\in\left\{\frac{1}{2},1,\cdots,R\right\}$，有

$$\left\|\nabla\mathcal{L}_{\langle t,r_1\rangle}^n\left(D^n;\theta_{\langle t,r_1\rangle}^n\right)-\nabla\mathcal{L}_{\langle t,r_2\rangle}^n\left(D^n;\theta_{\langle t,r_2\rangle}^n\right)\right\|_2\leqslant L_1\left\|\theta_{\langle t,r_1\rangle}^n-\theta_{\langle t,r_2\rangle}^n\right\|_2。\tag{13}$$

假设 2. 无偏梯度与有界方差。每个客户端的本地模型在小批量数据B上的随机梯度$g_t^n=\nabla\mathcal{L}_t^n(\boldsymbol{\omega}_t^n;B_t^n)$是无偏的，满足

$$E_{B_t^n\subseteq D^n}\left[g_t^n\right]=\nabla\mathcal{L}_t^n\left(\boldsymbol{\omega}_t^n\right),\tag{14}$$

并且该随机梯度的方差有界，满足

$$E_{B_t^n\subseteq D^n}\left[\left\|\nabla\mathcal{L}_t^n\left(\boldsymbol{\omega}_t^n;B_t^n\right)-\nabla\mathcal{L}_t^n\left(\boldsymbol{\omega}_t^n\right)\right\|_2^2\right]\leqslant\sigma^2。\tag{15}$$

假设 3. 聚合收敛性。这是一个强假设，仅在定理2中使用。用于保证聚合后的模型期望损失始终不高于聚合前各客户端期望损失的加权平均。两者关系为

$$E\left[\mathcal{L}_{\text{cla},\left\langle t+1,\frac{1}{2}\right\rangle}\right]\leqslant\frac{1}{N}\sum_{n=1}^{N}E\left[\mathcal{L}_{\text{cla},\langle t,R\rangle}^n\right]。\tag{16}$$

定理 1. 正则项有界。正则项的本质是2个行元素之和为一矩阵中在对应位置上的差值的平方之和。$\mathcal{L}_{\text{reg}}^n$的界为

$$0<\mathcal{L}_{\text{reg}}^n<\frac{2}{|C|}。\tag{17}$$

定理 2. 1轮偏差。基于上述3个假设和定理1，对于任意客户端在每一轮通信后，满足

$$E\left[\mathcal{L}_{\left\langle t+1,\frac{1}{2}\right\rangle}\right]\leqslant E\left[\mathcal{L}_{\left\langle t,\frac{1}{2}\right\rangle}\right]-\left(\eta-\frac{L_1\eta^2}{2}\right)\sum_{r=\frac{1}{2}}^{R}\left\|\nabla\mathcal{L}_{\langle t,r\rangle}\right\|_2^2+\frac{RL_1\eta^2\sigma^2}{2}+\frac{2\mu}{|c|}。\tag{18}$$

此处证明参考文献[11]。定理2展示了客户端在1轮通信后其本地目标函数的偏差界限。

定理 3. FedDW在non-IID数据下的收敛速率。基于上述3个假设与定理1、定理2，$\mathcal{L}^*$记为本地最优解。对于任意客户端，给定任意$\varepsilon>0$，有

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{r=\frac{1}{2}}^{R}E\left[\left\|\nabla\mathcal{L}_{\langle t,r\rangle}^n\right\|_2^2\right]\leqslant\frac{2\left(\mathcal{L}_{\left\langle 1,\frac{1}{2}\right\rangle}^n-\mathcal{L}^*\right)}{T(2\eta-L_1\eta^2)}+\frac{L_1R\eta^2\sigma^2+4\mu}{(2\eta-L_1\eta^2)|C|}\leqslant\varepsilon。\tag{19}$$

同时满足

$$T\leqslant\frac{2|C|\left(\mathcal{L}_{\left\langle 1,\frac{1}{2}\right\rangle}^n-\mathcal{L}^*\right)}{\varepsilon|C|(2\eta-L_1\eta^2)-|C|L_1(R\eta^2\sigma^2)-4\mu},\tag{20}$$

以及

$$\eta<\frac{\varepsilon|C|+\sqrt{\varepsilon^2|C|^2-4\mu L_1|C|(\varepsilon-R\sigma^2)}}{L_1|C|(\varepsilon-R\sigma^2)}。\tag{21}$$

因此，每个客户端的本地模型可以以非凸优化收敛率$\varepsilon\sim O\left(\frac{1}{T}\right)$收敛。定理证明参考文献[9]。

3.2 正则化项属性

下面分析正则项自身的性质。为便于分析，下文将项$\sigma(\boldsymbol{\omega}_i^n(\boldsymbol{\omega}_i^n)^{\mathrm{T}})$转换为矩阵$\boldsymbol{A}$，满足

$$\boldsymbol{A}=\{\boldsymbol{A}\in\mathbb{R}^{|C|\times|C|}|0\leqslant A_{ij},\sum_{j=1}^{n}A_{ij}=1,\forall i,j\}。\tag{22}$$

同时，将$\tilde{\boldsymbol{\Omega}}_t^{\text{global}}$简记为$\boldsymbol{\Omega}$，因此正则项$\mathcal{L}_{\text{reg}}$的计算公式可转化$\mathcal{L}_{\text{reg}}(\boldsymbol{A})=\|\boldsymbol{\Omega}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\|_{\mathrm{F}}^2$。下面本文给出$\mathcal{L}_{\text{reg}}(\boldsymbol{A})$的1阶导数以及2阶导数对应的Hessian矩阵$\boldsymbol{H}(\mathcal{L}_{\text{reg}})$：

$$\nabla\mathcal{L}_{\text{reg}}(\boldsymbol{A})=-2(\boldsymbol{\Omega}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})\boldsymbol{A},\tag{23}$$

$$\boldsymbol{H}(\mathcal{L}_{\text{reg}})=2(\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\otimes\boldsymbol{I})+2(\boldsymbol{I}\otimes\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}})-2\boldsymbol{I}\otimes(\boldsymbol{\Omega}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}),\tag{24}$$

其中$\boldsymbol{I}$是单位矩阵，$\otimes$表示Kronecker积。可以观察到，$\nabla\mathcal{L}_{\text{reg}}(\boldsymbol{A})$仅涉及神经网络最后1层的参数矩阵，与其他参数和网络结构无关。那么在反向传播过程中，正则项所产生的梯度传播只会从损失值流到最后1层参数，与其他从参数无关。因此正则项的计算开销不会随模型大小而改变。同时其梯度对任意数据保持不变，FedDW在1个batch中只需计算1次梯度。Hessian矩阵$\boldsymbol{H}(\mathcal{L}_{\text{reg}})$中包含正项与负项，表明它不是正定的。因此，$\mathcal{L}_{\text{reg}}$不是一个凸函数。尽管先前的收敛性分析中未要求凸性，但一个凸的$\mathcal{L}_{\text{reg}}$会对模型训练更有利。在实践中，可以通过线性近似$\mathcal{L}_{\text{reg}}$构造凸函数。使用$\boldsymbol{A}_0$作为线性化的参考点，对$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$进行泰勒展开：$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\approx\boldsymbol{A}_0^{\mathrm{T}}\boldsymbol{A}_0+\boldsymbol{A}_0^{\mathrm{T}}(\boldsymbol{A}-\boldsymbol{A}_0)+(\boldsymbol{A}-\boldsymbol{A}_0)^{\mathrm{T}}\boldsymbol{A}_0$，将其代入正则项计算公式中，得到

$$\mathcal{L}_{\text{reg}}(\boldsymbol{A})\approx\|\boldsymbol{\Omega}-\boldsymbol{A}_0^{\mathrm{T}}\boldsymbol{A}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}_0+\boldsymbol{A}_0^{\mathrm{T}}\boldsymbol{A}_0\|_{\mathrm{F}}^2。\tag{25}$$

由于$\boldsymbol{A}_0$是初始矩阵且不被更新，近似函数变成关于$\boldsymbol{A}$的二次凸函数。在迭代优化过程中，可定期更新参考点$\boldsymbol{A}_0$以逼近最优点。

4 实　　验

4.1 实验设置

本文将FedDW与10个主流的联邦学习框架进

行了对比, 包括 FedAvg[3], FedUV[35], MOON[22], FedBABU[36], FedProto[11], FedProx[21], FedRep[26], FedALA[27], Per-FedAvg[37], FedDistill+(在 Fedistill[14] 的基础上添加了整个网络的全局聚合)。其中 FedUV 和本文思想类似, 都是通过正则模型参数分布来缓解数据异构。Fedistill+和 FedDW 一样, 主要使用软标签作为 DLE 数据。MOON, FedProx, FedALA 是主流的以神经网络参数为 DLE 数据的方法。FedProto 是主流的以特征向量为 DLE 数据的方法。FedBABU 和 FedRep 是将模型分为特征提取器和分类器的奠基性工作而 Per-FedAvg 是用深度元学习的思想解决异构问题。在计算机视觉(CV)领域, 本文使用 MNIST, CIFAR-10, CIFAR-100 数据集; 在自然语言处理(NLP)领域, 使用 IMDB 数据集。每个数据集使用不同的模型: MNIST 使用 2 层卷积神经网络, CIFAR-10 使用带最大池化的 CNN, CIFAR-100 使用 ResNet-18, IMDB 使用双向 LSTM。

本文所有实验的神经网络模型的映射层 f_π 均为 2 层的全连接网络, 分类层 f_ω 为 1 层全连接层, 输入维度均为 128。所有实验均基于 PyTorch[38] 框架在 8 张 NVIDIA RTX4090 组成的服务器上运行。为保证公平性, 防止针对某一特定数据集进行特定过分调参。所有实验均采用 Adam[39] 优化器, 学习率设为 0.001, 其他参数均为 PyTorch 默认设置。其他框架的超参数均参考其原始论文。批大小在 CV 任务中设为 128, 在 NLP 中设为 512, 所有结果取 5 次实验的平均值。FedDW 的分类层去除了偏置项, 此操作仅适用于 FedDW, 其他对比方法的分类层仍为标准的线性层。为了模拟异构数据分布, 本文采用常用的 Dirichlet 分布 $Dir(\beta)$ 来构造 non-IID 的数据分布, 其中参数 β 控制数据异构性的程度, 参数 β 越小, 数据的异构性越强。

4.2 性能对比

本文设计了实际异构环境(β=0.5, 用户参与率为 1.0)和极端异构环境(β=0.1, 用户参与率为 0.5)2 种异构环境进行对比。在 CV 领域的实验设置中, 本文将每轮联邦通信定义为包含 50 次的全局通信轮次, 并设定每个客户端在本地执行 5 个训练轮次后再进行 1 次模型聚合操作。相较而言, 在 NLP 领域的实验中, 考虑到文本数据的特性以及计算资源的分配, 我们对 IMDB 数据集的联邦训练过程仅进行了 10 轮通信。一个实用的联邦学习框架应在尽可能少的通信轮数下保持较高准确率, 而不仅仅追求最终准确率。实验结果如表 1 所示, FedDW 在所有设定下始终表现出最佳或竞争性的性能。尽管在异构性增强时, 整体准确率有所下降, 但 FedDW 表现出最小的性能退化: 在 CIFAR-10 上下降了 8.51 个百分点, 在 CIFAR-100 上下降了 13.32 个百分点。在收敛速度方面, FedDW 不仅实现了竞争力的性能, 还展现了最快的收敛速度, 如图 5 所示。

4.3 可扩展性

为验证 FedDW 框架的可扩展性, 本文在 CIFAR-

Table 1 Classification Accuracy of Different Methods

表 1 不同方法的分类准确率 %

方法	实际异构环境				极端异构环境			
	MNIST	CIFAR-10	CIFAR-100	IMDB	MNIST	CIFAR-10	CIFAR-100	IMDB
FedAvg	99.31±0.2	63.44±0.4	62.16±0.6	83.33±0.4	95.05±0.5	48.36±0.3	43.47±0.5	71.09±0.9
MOON	99.42±0.1	64.33±0.5	63.35±0.3	81.14±0.3	96.11±0.1	54.01±0.4	49.04±0.2	80.28±0.4
FedBABU	99.51±0.1	65.59±0.2	63.49±0.2	82.41±0.4	**96.78±0.2**	53.06±0.2	50.12±0.3	81.08±0.2
FedUV	**99.52±0.2**	65.05±0.3	63.35±0.2	68.22±0.6	96.65±0.3	54.97±0.3	44.09±0.5	55.12±0.4
FedProx	99.39±0.1	64.74±0.2	62.81±1.0	73.12±0.2	95.12±0.4	51.39±0.3	46.07±0.2	66.84±0.4
FedRep	96.88±0.3	52.42±1.1	39.03±2.2	74.64±1.2	67.23±1.2	28.05±1.5	23.46±0.9	54.13±0.8
FedProto	97.11±0.1	45.52±0.8	23.15±0.9	70.69±0.8	49.83±4.2	27.64±1.6	16.48±1.3	53.85±2.4
FedALA	99.02±0.1	48.39±1.5	41.97±3.3	74.14±0.8	79.16±1.2	32.58±0.8	28.16±1.2	61.46±0.7
PerFedAvg	99.12±0.1	64.23±0.1	62.22±0.5	79.76±0.6	96.21±0.5	–	–	63.45±0.8
FedDistill+	99.05±0.0	63.89±1.1	63.11±1.2	80.64±1.6	96.09±2.1	53.11±0.9	47.17±1.7	64.08±1.1
Local	95.35±3.2	42.38±5.1	22.83±6.2	63.18±3.6	55.76±9.2	25.75±3.2	15.54±8.8	51.11±4.5
FedDW(本文)	**99.52±0.1**	**66.24±0.6**	**63.53±0.3**	**84.46±0.3**	96.66±0.5	**57.73±0.5**	**50.21±0.3**	**82.81±0.4**

注: Local 表示本地训练不进行联邦学习。黑体数值表示最佳结果; "–"表示模型训练崩溃; "±"前后的数值分别表示准确率和多次实验所得到的标准差。

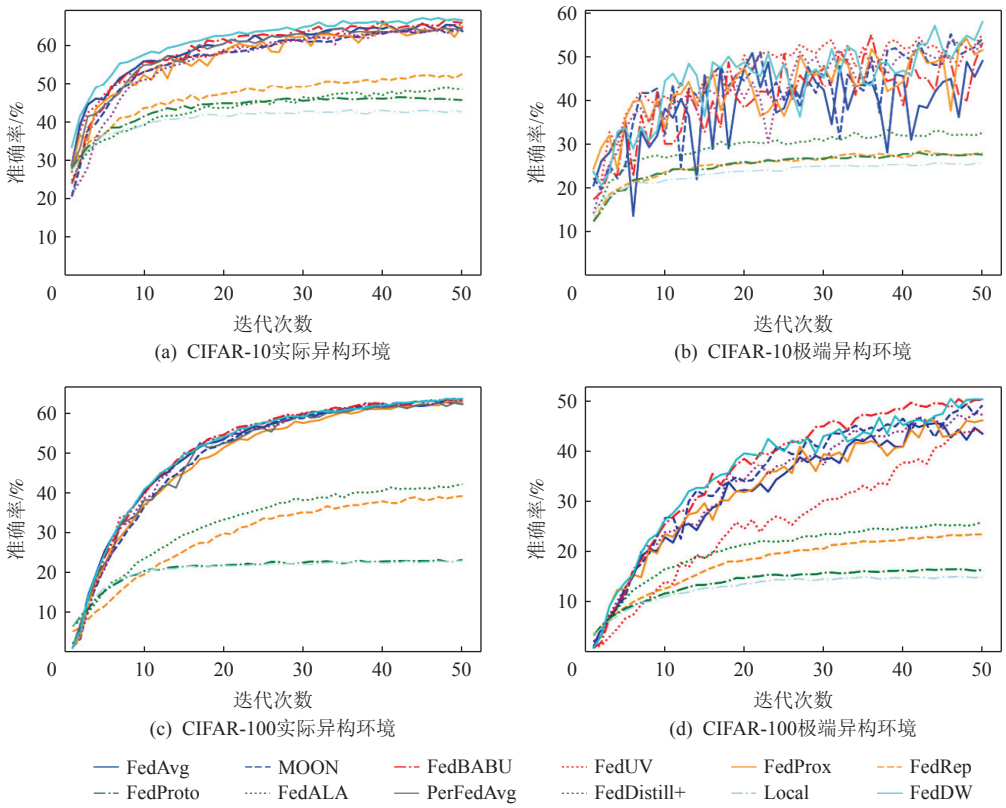


Fig. 5 Learning curves of different FL methods in different environments

图 5 不同的联邦学习方法在不同环境下的学习曲线

10 数据集上进行了客户端数量和训练轮次的可扩展性实验(在实际异构环境中)。同时，在 CIFAR-100 数据集上进行了模型可扩展性实验(在实际异构环境中)。实验结果如表 2 所示。

1)客户端数量可扩展性。在该实验中改变参与训练的客户端数量。在上文的性能对比实验中客户端数量为 10。本实验将其增加至 50 和 100。观察到随着客户端数量增加，所有方法的模型准确率下降，这是可以预期的，因为总数据量不变，每个客户端分配的数据减少，导致模型泛化能力下降。但即使在

Table 2 Accuracy of Scalability Experiment

表 2 扩展性实验的准确率 %

方法	客户端数量可扩展性			训练轮次可扩展性		模型可扩展性		
	N=10	N=50	N=100	$S_r=25$，$C_r=10$	$S_r=125$，$C_r=2$	ResNet-18	ShuffleNet	MobileNet
FedAvg	63.44±0.4	56.07±0.1	51.41±0.3	65.72±0.3	66.09±0.2	62.16±0.6	50.41±0.4	52.77±0.5
MOON	64.33±0.5	53.59±0.4	49.59±0.5	65.84±0.4	65.56±0.7	63.35±0.3	50.87±0.3	52.98±0.5
FedBABU	65.59±0.2	56.12±0.2	50.55±0.1	66.89±0.1	66.33±0.2	63.49±0.2	**52.08±0.2**	53.05±0.4
FedUV	65.05±0.3	56.81±0.2	–	66.81±0.3	67.01±0.4	63.35±0.2	51.97±0.2	53.13±0.1
FedProx	64.74±0.2	55.87±0.6	51.19±0.5	65.84±0.2	66.19±0.4	62.81±1.0	50.34±0.3	50.52±0.2
FedRep	52.42±1.1	38.66±1.3	35.12±0.9	49.89±0.7	50.11±0.5	39.03±2.2	40.56±0.7	41.66±0.9
FedProto	45.52±0.8	37.15±1.1	34.57±0.8	44.52±0.6	45.04±0.3	23.15±0.9	28.34±1.4	30.43±0.8
FedALA	48.39±1.5	39.71±0.7	35.87±0.8	46.65±1.3	47.54±2.4	41.97±3.3	34.17±0.4	39.44±0.5
PerFedAvg	64.23±0.1	55.23±0.3	50.11±0.3	63.55±0.2	64.53±0.5	62.22±0.5	48.44±0.3	49.11±0.3
FedDistill+	63.89±1.1	54.94±2.1	50.78±0.5	63.48±1.3	66.14±0.3	63.11±1.2	51.11±0.4	52.88±0.5
Local	42.38±5.1	34.31±1.3	28.13±1.2	40.68±3.4	41.02±2.8	22.83±6.2	25.71±1.2	27.94±0.5
FedDW（本文）	**66.24±0.6**	**57.6±0.3**	**51.73±0.4**	**67.06±0.4**	**67.21±0.2**	**63.53±0.3**	51.92±0.4	**53.71±0.3**

注：Local 表示本地训练不进行联邦学习。黑体数值表示最佳结果；“–”表示模型训练崩溃；“±”前后数值分别为准确率和多次实验所得到的标准差。

客户端数量增加多的情况下，FedDW 依然取得了最高准确率。

2)训练轮次可扩展性。在上文的性能对比实验中本地训练轮次C_r=5，通信轮数为S_r=50，总训练轮次为 250。本实验通过保持总训练轮次不变，动态调整C_r和S_r来测试 2 个极端情况：$S_r=25$，$C_r=10$与$S_r=125$，$C_r=2$。实验结果如表 2 所示，在这 2 种情况下，FedDW 均取得最佳效果。两者间的准确率差异仅为 0.14 个百分点。与 FedDW 所对比的 10 种方法相比，该差值是最小的。

3)模型可扩展性。在该实验中，本文在 CIFAR-100 数据集上测试了 ShuffleNet[40] 和 MobileNet[41] 两个 CNN 模型并与上文的性能对比实验中的 ResNet-18 模型进行比较。实验结果表明，FedDW 在 MobileNet 上取得了最佳性能，在 ShuffleNet 上排名第 2，仅比表现最好的方法 FedBABU 差 0.16 个百分点。结合表 1 中 FedDW 在 IMDB 数据集(使用 LSTM)的结果，FedDW 展现出优秀的模型可扩展性。

为进一步展现 FedDW 在模型上的可扩展性，将其应用到基于 Transformer 架构的网络模型。使用 vision Transformer[42] 的 base 版本在实际异构环境的环境下的 CIFAR-100 数据集上训练 25 轮，一共实验 5 次，取平均值，训练结果如表 3 所示。可以发现，FedDW 在使用更大且基于 Transformer 的神经网络模型的条件之下依然有效，而且性能得到了非常显著的提升。而所对比的部分方法，甚至出现了性能下降的情况。

Table 3 Accuracy Comparison of Different Methods on Transformer Based Architecture Models

表 3 不同方法在基于 Transformer 架构模型上的准确率对比 %

方法	准确率	方法	准确率
FedAvg	62.07±0.21	FedProto	22.39±4.12
MOON	53.12±0.6	FedALA	45.52±1.34
FedBABU	56.84±0.87	PerFedAvg	44.32±0.87
FedUV	66.84±0.5	FedDistill+	63.96±1.43
FedProx	74.7±0.3	Local	19.98±0.56
FedRep	49.21±2.3	FedDW（本文）	**79.62±0.6**

注：Local 表示本地训练不进行联邦学习。黑体数值表示最佳结果；“±”前后的数值分别表示准确率和多次实验所得到的标准差。

4.4 效 率

联邦学习框架的效率主要体现在通信效率和本地计算效率 2 个方面。在 FedDW 中，基于 FedAvg 每个客户端除了上传模型参数外，还需额外向服务器共享 SL 矩阵及每类样本数量。这部分带来的通信开销为$2|C|+2|C|^2$，相较于完整神经网络模型的通信量来说是可以忽略的。FedDW 的正则项仅涉及分类层参数，其反向传播构建的计算图不会延伸至其他网络参数，从而有效减少了本地训练过程中的计算复杂度。在表 4 中列出了在 CIFAR-100 数据集上使用 ResNet-18 模型进行本地训练时的耗时情况。对比结果显示，FedDW 的训练时间最短，在所有方法中执行效率最高，验证了其在实际部署场景中的优势。

Table 4 Comparative Experiment of Time-Consuming

表 4 耗时对比实验

	FedAvg	FedProto	FedDW（本文）	FedBABU	MOON	FedRep
耗时倍数	1	1.01	0.953	0.96	1.13	0.99
	FedUV	FedProx	FedALA	PerFedAvg	FedPer	FedDistill+
耗时倍数	0.974	1.03	0.975	1.11	1.06	1.02

4.5 兼容性和可视化

FedDW 的正则项仅作用于分类层参数，这种设计提升了和其他联邦优化方法的兼容性，减少了梯度冲突和优化目标之间的干扰。将 FedDW 的优化策略抽象为“DW”，并将其与多种主流联邦学习方法组合，评估其整体性能提升情况。如表 5 所示，DW 与大多数方法的组合均带来了准确率的提升，验证了 DW 的普适性和通用优化能力。唯一例外是与 FedUV 的组合未表现出性能提升。其原因在于 FedUV 通过调整神经网络参数分布以保持在 non-IID 数据上有 IID 特性。这与 DW 虽方法上不同，但目标一致。因此两者结合不会进一步提升准确率。

Table 5 Accuracy Comparison of Single Method and Combined Method

表 5 单个方法和组合方法的准确率比较 %

方法	单个方法	结合 DW
FedDW（本文）	50.21±0.3	
FedDistill+	43.96±0.8	47.26±0.6
MOON	49.04±0.9	52.81±0.2
FedUV	44.09±0.5	44.09±0.8
FedProx	46.07±0.3	48.39±0.8

注：“±”前数值表示平均值，“±”后数值表示标准差。

本文对在 3 种不同实验条件下，使用 MNIST 数据集进行训练所生成的 SL 矩阵和 CR 矩阵进行了详细的可视化分析，以便更清晰地展示各条件下的矩阵变化和特征。如图 6~8 的热力图所示，在 IID 条件下，FedAvg 的 SL 与 CR 矩阵高度相似。non-IID 条件

下，两者相似性大幅下降。然而，在 non-IID 条件下，FedDW 的 SL 矩阵依然与 IID 情况下相似，并且 CR 矩阵也被有效正则而与 SL 矩阵保持一致性。这说明 SL 矩阵在 FedDW 中具备良好的全局泛化能力，即便在数据高度异质的情况下，也能有效引导 CR 矩阵对齐。从而缓解数据分布不一致带来的性能损失。这展现出 FedDW 在异构场景下的强大鲁棒性与正则能力。同时可以注意到，即使在 non-IID 环境下生成的 SL 矩阵其结构仍与 IID 环境下所得到的 SL 矩阵高度一致。这表明 SL 矩阵相比于 CR 矩阵展现出更强的鲁棒性。因此本文将 SL 矩阵视为全局共享信息而非 CR 矩阵具有充分的合理性。

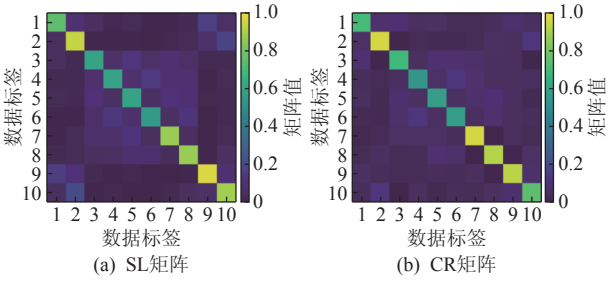


Fig. 6 Thermal map of SL matrix and CR matrix of FedAvg in IID environment

图 6 FedAvg 在 IID 环境下的 SL 矩阵和 CR 矩阵热力图

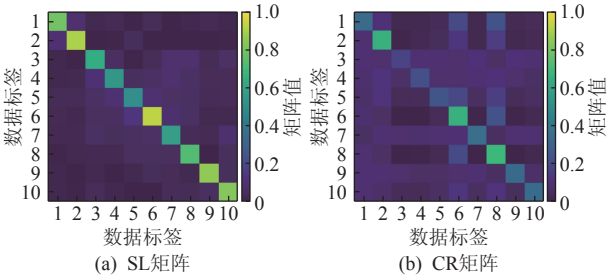


Fig. 7 Thermal map of SL matrix and CR matrix of Fedavg in non-IID environment

图 7 FedAvg 在 non-IID 环境下的 SL 矩阵和 CR 矩阵热力图

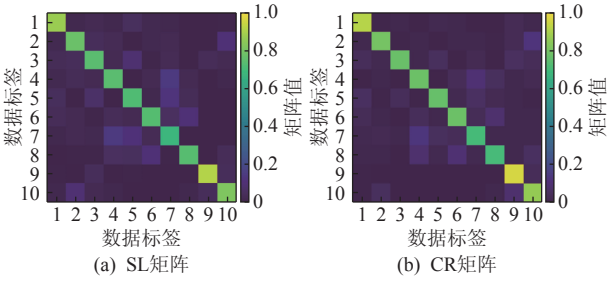


Fig. 8 Thermal map of SL matrix and CR matrix of FedDW in non-IID environment

图 8 FedDW 在 non-IID 环境下的 SL 矩阵和 CR 矩阵热力图

5 结 论

FedDW 是一种简单而有效的联邦学习方法，它以最小的计算和通信开销解决了数据异构问题。广泛的可扩展性实验证明了它在各种客户端数量、训练配置和模型架构上的鲁棒性和适应性。另一方面，也通过理论分析和可视化实验证明了 FedDW 的合理性。值得注意的是，FedDW 中的正则项仅限于分类层，使其计算成本与模型大小无关。而这样的性质则非常适合大规模网络。然而，由于目前的实验条件限制，它在真正大规模联邦学习场景中的性能仍未得到探索。未来的工作将侧重于在这种环境中评估 FedDW，并将其应用扩展到更广泛的深度学习任务，包括视觉、语言和多模态学习领域。

作者贡献声明：刘佳瑜提出论文主体思想，完成实验并撰写论文；王勇负责论文构思，指导修改论文；杨静指导并修改论文；王念滨指导并修改论文。

参 考 文 献

[1] Zhang Kaiyue, Song Xuan, Zhang Chenhan, et al. Challenges and future directions of secure federated learning: A survey[J]. Frontiers of Computer Science, 2022, 16(5): 1−8

[2] Lin Weiwei, Shi Fang, Zeng Lan, et al. A review of federated learning open-source frameworks[J]. Journal of Computer Research and Development, 2023, 60(7): 1551−1580 (in Chinese)
(林伟伟, 石方, 曾岚, 等. 联邦学习开源框架综述[J]. 计算机研究与发展, 2023, 60(7): 1551−1580)

[3] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Proc of the 20th Int Conf on Artificial intelligence and statistics. New York: PMLR, 2017: 1273−1282

[4] Guo Kuangpu, Ding Yuhe, Liang Jian, et al. Not all minorities are equal: Empty-class-aware distillation for heterogeneous federated Learning[J]. arXiv preprint, arXiv: 2401.02329, 2024

[5] Ye Mang, Fang Xiuwen, Du Bo, et al. Heterogeneous federated learning: State-of-the-art and research challenges[J]. ACM Computing Surveys, 2023, 56(3): 1−44

[6] Zhao Yue, Li Meng, Lai Liangzhen, et al. Federated learning with non-iid data[J]. arXiv preprint, arXiv: 1806.00582, 2018

[7] Shi Jiangming, Zheng Shanshan, Yin Xiangbo, et al. Clip-guided federated learning on heterogeneous and long-tailed data[J]. arXiv preprint, arXiv: 2312.08648, 2023

[8] Kang Bingyi, Xie Saining, Rohrbach M, et al. Decoupling representation and classifier for long-tailed recognition[J]. arXiv preprint, arXiv: 1910.09217, 2019

[9] Yi Liping, Wang Gang, Liu Xiaoguang, et al. FedGH: Heterogeneous federated learning with generalized global header[C]//Proc of the 31st ACM Int Conf on Multimedia. New York: ACM, 2023: 8686−8696

[10] Zhang Jianqing, Liu Yang, Hua Yang, et al. FedTGP: Trainable global prototypes with adaptive-margin-enhanced contrastive learning for data and model heterogeneity in federated learning[C]//Proc of the 38th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2024: 16768−16776

[11] Tan Yue, Long Guodong, Liu Lu, et al. FedProto: Federated prototype learning across heterogeneous clients[C]//Proc of the 36th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2022: 8432−8440

[12] Xie Chulin, Huang D, Chu Wenda, et al. PerAda: Parameter-efficient federated learning personalization with generalization guarantees[C]//Proc of the 37th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2024: 23838−23848

[13] Xie Luyuan, Lin Manqing, Luan Tianyu, et al. MH-pFLID: Model heterogeneous personalized federated learning via injection and distillation for medical data analysis[J]. arXiv preprint, arXiv: 2405.06822, 2024

[14] 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 preprint, arXiv: 1811.11479, 2018

[15] Hinton G, Vinyals O, Dean J, et al. Distilling the knowledge in a neural network[J]. arXiv preprint, arXiv: 1503.02531, 2015

[16] Zhu Ligeng, Liu Zhijian, Han Song. Deep leakage from gradients[C]//Proc of the 33rd Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2019: 14747−14756

[17] He Kaiming, Fan Haoqi, Wu Yuxin, et al. Momentum contrast for unsupervised visual representation learning[C]//Proc of the 33rd IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2020: 9729−9738

[18] Yang Yuhao, Huang Chao, Xia Lianghao, et al. Debiased contrastive learning for sequential recommendation[C]//Proc of the 32nd Int Conf on World Wide Web. New York: ACM, 2023: 1063−1073

[19] Qian Zhongsheng, Huang Heng, Zhu Hui, et al. Multiview graph contrastive learning recommendation method with fusion-layer attention mechanism[J]. Journal of Computer Research and Development, 2025, 62(1): 160−178 (in Chinese)
(钱忠胜, 黄恒, 朱辉, 等. 融合层注意力机制的多视角图对比学习推荐方法[J]. 计算机研究与发展, 2025, 62(1): 160−178)

[20] Zhang Jianqing, Liu Yang, Hua Yang, et al. An upload-efficient scheme for transferring knowledge from a server-side pre-trained generator to clients in heterogeneous federated learning[C]//Proc of the 37th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2024: 12109−12119

[21] Li Tian, Sahu A K, Zaheer M, et al. Federated optimization in heterogeneous networks[J]. Machine Learning and Systems, 2020, 2(1): 429−450

[22] Li Qinbin, He Bingsheng, Song D. Model-contrastive federated learning[C]//Proc of the 34th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2021: 10713−10722

[23] Durmus A E, Yue Z, Ramon M, et al. Federated learning based on dynamic regularization[J]. arXiv preprint, arXiv: 2111.04263, 2021

[24] Chen Yuhang, Huang Wenke, Ye Mang. Fair federated learning under domain skew with local consistency and domain diversity[C]//Proc of the 34th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2024: 12077−12086

[25] Sattler F, Korjakow T, Rischke R, et al. FedAUX: Leveraging unlabeled auxiliary data in federated learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 34(9): 5531−5543

[26] Collins L, Hassani H, Mokhtari A, et al. Exploiting shared representations for personalized federated learning[C]//Proc of the 38th Int Conf on Machine Learning. New York: ICML, 2021: 2089−2099

[27] Zhang Jianqing, Hua Yua, Wang Hao, et al. FedALA: Adaptive local aggregation for personalized federated learning[C]//Proc of the 36th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2023: 11237−11244

[28] Arivazhagan M G, Aggarwal V, Singh A K, et al. Federated learning with personalization layers[J]. arXiv preprint, arXiv: 1912.00818, 2019

[29] Gao Yujia, Wang Pengfei, Liu Liang, et al. Personalized federated learning method based on attention-enhanced meta-learning network[J]. Journal of Computer Research and Development, 2024, 61(1): 196−208 (in Chinese)
(高雨佳, 王鹏飞, 刘亮, 等. 基于注意力增强元学习网络的个性化联邦学习方法[J]. 计算机研究与发展, 2024, 61(1): 196−208)

[30] Tan A, Yu Han, Cui Lizhen, et al. Towards personalized federated learning[J]. IEEE Transactions on Neural Networks and Learning Systems, 2022, 34(12): 9587−9603

[31] Shi Hongzhi, Zhao Jian, Zhao Yaqian, et al. Survey on system optimization for mixture of experts in the era of large models[J]. Journal of Computer Research and Development, 2025, 62(5): 1164−1189 (in Chinese)
(史宏志, 赵健, 赵雅倩, 等. 大模型时代的混合专家系统优化综述[J]. 计算机研究与发展, 2025, 62(5): 1164−1189)

[32] Tai Jianwei, Yang Shuangning, Wang Jiajia, et al. A review of adversarial attacks and defenses on large language models[J]. Journal of Computer Research and Development, 2025, 62(3): 563−588 (in Chinese)
(台建玮, 杨双宁, 王佳佳, 等. 大语言模型对抗性攻击与防御综述[J]. 计算机研究与发展, 2025, 62(3): 563−588)

[33] Chen T, Kornblith S, Norouzi M, et al. A simple framework for contrastive learning of visual representations[C]//Proc of the 37th Int Conf on Machine Learning. New York: ICML, 2020: 1597−1607

[34] Abadi M, Chu A, Goodfellow I, et al. Deep learning with differential privacy[C]//Proc of the 23rd ACM SIGSAC Conf on Computer and Communications Security. New York: ACM, 2016: 308−318

[35] Son H M, Kim M H, Chung T M, et al. FedUV: Uniformity and variance for heterogeneous federated learning[C]//Proc of the 34th IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2024: 5863−5872

[36] Oh J, Kim S, Yun S Y. FedBABU: Towards enhanced representation for federated image classification[J]. arXiv preprint, arXiv: 2106.06042, 2021

[37] Fallah A, Mokhtari A, Ozdaglar A. Personalized federated learning: A

meta-learning approach[J]. arXiv preprint, arXiv: 2002.07948, 2020

[38] Paszke A, Gross S, Massa F, et al. PyTorch: An imperative style, high-performance deep learning library[C]//Proc of the 33rd Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2019: 8024−8035

[39] Kingma D P, Ba J L. Adam: A method for stochastic optimization[J]. arXiv preprint, arXiv: 1412.6980, 2014

[40] Zhang Xiangyu, Zhou Xinyu, Lin Mengxiao, et al. ShuffleNet: An extremely efficient convolutional neural network for mobile devices[C]//Proc of the 31st IEEE/CVF Conf on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2018: 6848−6856

[41] Howard A G, Zhu Menglong, Chen Bo, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint, arXiv: 1704.04861, 2017

[42] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16×16 words: Transformers for image recognition at scale[J]. arXiv preprint, arXiv: 2010.11929, 2020



Liu Jiayu, born in 2000. PhD candidate. Student member of CCF. His main research interests include federated learning, reinforcement learning, and expert system of LLM mixture.

刘佳瑜, 2000 年生。博士研究生。CCF 学生会员。主要研究方向为联邦学习、强化学习、大语言模型混合专家系统。



Wang Yong, born in 1983. PhD, associate professor. Senior member of CCF. His main research interests include artificial intelligence, privacy computing, and Internet of things.

王　勇, 1983 年生。博士，副教授。CCF 高级会员。主要研究方向为人工智能、隐私计算、物联网。



Yang Jing, born in 1962. PhD, professor. Senior member of CCF. Her main research interests include artificial intelligence, social computing, and privacy protection.

杨　静, 1962 年生。博士，教授。CCF 高级会员。主要研究方向为人工智能、社会计算、隐私保护。



Wang Nianbin, born in 1967. PhD, professor. Senior member of CCF. His main research interests include artificial intelligence, large language model, and data mining.

王念滨, 1967 年生。博士，教授。CCF 高级会员。主要研究方向为人工智能、大语言模型、数据挖掘。