小 型 微 型 计 算 机 系 统
Journal of Chinese Computer Systems
DOI:10.20009/j.cnki.21-1106/TP.2025-0233
2026 年 5 月 第 5 期
Vol.47 No.5 2026

通信高效的自适应联邦剪枝优化方法

裴锡凯[1,2],王柯阳[1],周 潼[1],张凤荔[1],王瑞锦[1]

[1](电子科技大学 信息与软件工程学院,成都 610051)
[2](成都民航空管科技发展有限公司,成都 610041)
E-mail:ruijinwang@uestc.edu

摘 要:联邦学习中深度神经网络的参数量巨大,每轮训练客户端需上传完整模型更新参数,在带宽受限环境下,通信开销成为系统性能瓶颈,尤其是在带宽受限环境下.因此,在保证模型性能的同时降低通信开销是联邦学习研究的关键问题之一.针对上述挑战,本文提出了一种通信高效的自适应联邦剪枝优化方法(communication-efficient adaptive federated pruning optimization method,CEAFL).核心是阶段式自适应模型剪枝算法,分为初始剪枝和自适应剪枝两个阶段,用梯度重要性进行模型剪枝,实现轻量化传输.此外,设计了集成分类器复用的模型微调算法,提升泛化能力和数据分布感知能力.实验表明,相较于基准方法,该方法在多个数据集上的模型精度提升了超过 0.5%,同时通信量减少了约 38%,展现了其在实际应用中的潜力.

关 键 词:联邦学习;模型压缩;模型裁剪;知识蒸馏
中图分类号:TP181 **文献标识码**:A **文 章 编 号**:1000-1220(2026)05-1225-11

Adaptive Federated Pruning Optimization Method with Efficient Communication

PEI Xikai[1,2],WANG Keyang[1],ZHOU Tong[1],ZHANG Fengli[1],WANG Ruijing[1]
[1](School of Information and Software Engineering,University of Electronic Science and Technology of China,Chengdu 610051,China)
[2](Chengdu Civil Aviation Air Traffic Control Technology Development Co.,Ltd.,Chengdu 610041,China)

Abstract:The number of parameters of deep neural networks in federated learning is huge. The client needs to upload the complete model update for each round of training,which makes the communication overhead become the bottleneck of system performance,especially in bandwidth-constrained environments. Therefore,reducing communication while ensuring model performance is one of the key issues in federated learning research. In response to the above challenges,this paper proposes a communication-efficient adaptive federated pruning optimization method(CEAFL). The core is a staged adaptive model pruning algorithm,which is divided into two stages:initial pruning and adaptive pruning. The model is pruned using gradient importance to achieve lightweight transmission. In addition,a model fine-tuning algorithm for integrated classifier reuse is designed to improve generalization ability and data distribution perception ability. Experiments show that compared with the benchmark method,this method improves the model accuracy on multiple data sets by more than 0.5%,while reducing the communication volume by about 38%,demonstrating its potential in practical applications.

Keywords:federated learning;model compression;model pruning;knowledge distillation

0 引 言

随着互联网技术的飞速发展和智能客户端的普及,数据生产与处理正快速向分布式发展.在金融、医疗等领域,跨组织数据协作的需求日益增长,但数据隐私保护和安全性问题成为制约其发展的关键瓶颈.传统集中式深度学习需要将数据汇聚到中心服务器进行处理,这种方式不仅面临巨大的通信开销,更存在严重的数据隐私泄露风险.在此背景下,联邦学习作为一种新兴的分布式学习范式应运而生[1],通过允许多个参与方在不直接交换原始数据的情况下协作训练全局模型,有效解决数据隐私保护与跨组织协作的矛盾.

联邦学习已在多个领域展现出重要价值[2].在医疗领域,联邦学习使医院间能够共享疾病诊断模型而不泄露患者敏感数据;在金融风控中,银行可以通过联邦学习联合构建反欺诈模型,同时严格遵守客户隐私保护法规;智能交通系统则利用联邦学习整合跨区域交通流量数据,优化信号灯控制策略.此外,在推荐系统、工业物联网、环境监测等领域,联邦学习也表现出独特的优势.王瑞锦团队的研究表明,通过构建双重防御机制[3],联邦学习能够有效打破数据孤岛,在保障隐私安全的前提下实现跨域协作,成为新一代分布式学习范式的关键技术.

尽管联邦学习具有显著优势,其在实际应用中仍面临核心挑战:客户端在训练过程中需要频繁上传完整的模型更新参数,使得通信开销成为系统性能瓶颈,尤其在带宽受限环境

收稿日期:2025-06-09 收修改稿日期:2025-07-31 基金项目:国家自然科学基金项目(62271128,U2333207)资助;四川省科技计划重点研发项目(2022ZDZX0004,2025YFHZ0302)资助;四川省科技计划"揭榜挂帅"项目(2023YFG0374)资助;成都重点研发支持计划项目(2025-YF08-00128-GX,2025-YF12-00029-RC)资助. 作者简介:**裴锡凯**,男,1977 年生,博士研究生,高级工程师,研究方向为联邦学习及安全;**王柯阳**,男,2003 年生,硕士研究生,CCF 会员,研究方向为联邦学习及安全;**周 潼**,男,2000 年生,硕士研究生,研究方向为联邦学习及安全;**张凤荔**,女,1963 年生,博士,教授,CCF 会员,研究方向为信息安全;**王瑞锦**(通信作者),男,1980 年生,博士,副教授,CCF 会员,研究方向为联邦学习及安全等.

下更为突出. 此外,大量低重要性参数的重复传输不仅浪费带宽资源,还会因网络延迟造成客户端掉线,影响全局模型的收敛速度和最终性能. 如何在保证模型精度的前提下有效降低通信负担,成为当前联邦学习研究的核心问题.

当前联邦学习的优化方法主要存在三类技术路线:量化方法通过降低参数精度减少传输数据量,但会加剧非独立同分布数据下的梯度偏差;剪枝方法通过移除冗余参数压缩模型规模,但静态剪枝策略难以适应动态网络环境;异构知识蒸馏方法通过师生模型传递知识,但存在架构对齐复杂和额外计算开销等问题. 这些方法普遍面临动态适应性不足、知识迁移效率低和资源消耗失衡的共性挑战,制约着联邦学习在资源受限场景的实际应用.

因此,本文将充分发挥联邦学习的模型压缩优势,结合自适应剪枝与知识蒸馏技术,提出通信高效的自适应联邦剪枝优化方法 CEAFL(communication-efficient adaptive federated pruning optimization method),客户端通过梯度敏感的参数筛选实现本地模型轻量化,服务器负责全局模型重配置与知识蒸馏,共同完成模型的协同优化. 针对剪枝过程中的性能损失问题,本文创新性地采用集成分类器复用机制,通过构建虚拟教师模型实现知识迁移,确保模型在参数缩减后仍保持模型性能和泛化能力.

本文的主要贡献如下:

1)提出两阶段自适应剪枝机制,通过梯度敏感的参数重要性评估和周期性模型重配置,实现高效通信.

2)设计集成分类器复用策略,构建虚拟教师模型进行特征空间与决策边界双重知识蒸馏,有效缓解剪枝带来的性能损失.

3)在经典图像分类数据集上验证 CEAFL 的有效性,相比主流基线算法,在通信效率、模型精度和异构环境适应性等方面均展现出显著优势.

1 相关工作

针对联邦学习中模型参数规模过大导致通信负担和计算开销大的问题,学术界中近年来主要聚焦于模型剪枝和知识蒸馏这两类优化策略,在提升资源受限环境下训练效率的同时,也带来了模型精度保持和剪枝策略优化等新的挑战.

1.1 模型剪枝优化方法

模型剪枝作为联邦学习中降低通信开销的重要技术手段,主要通过去除神经网络中的冗余权重来减小模型规模. 近年来,研究者们针对联邦学习的特性提出了多种剪枝优化策略,主要包括以下几类方法:

自适应剪枝策略是当前研究的重点方向之一. 这类方法通过动态调整剪枝率,使剪枝过程能够适应不同的计算资源、通信环境和数据分布特点. Mohammad 等人开发的 SparseFL 模型[4]创造性地融合了稀疏约束与压缩感知方法,采用分布式参数修剪策略有效提升了联邦学习的传输性能. 该方案在参数优化阶段加入稀疏限制,并运用压缩感知算法降低数据传输量. 研究发现,在数据分布不均匀的情况下,该框架的全局修剪方法可能会误删某些关键参数,从而降低模型准确率,这表明其在处理异构数据时仍存在改进潜力.

边缘辅助的剪枝方法是另一个重要研究方向. 陈晓研究组开发的 ASEAFL 方法[5]有效利用边缘计算资源,借助边缘节点实现动态稀疏化处理. 该算法运用非规则剪枝技术,能够依据终端设备的实际性能和网络条件灵活调节剪枝强度. 测试结果显示,这种边缘计算辅助方案不仅能够明显降低终端设备的运算延迟和能耗,还可以大幅减少因资源限制导致的掉队设备数量,进而提高整体训练效果. 不过,该方法对边缘服务器的依赖可能增加系统部署成本.

分层剪枝优化也是研究者关注的重点. Wang 研究团队提出的 FedADP 方案[6]采用创新的分层自适应剪枝方法,在联邦学习过程中为神经网络各层动态确定最佳剪枝比例. 该技术通过评估不同网络层对模型性能的影响程度,实现了通信开销与模型准确性的优化平衡. 然而需要指出的是,该方法需要预先评估客户端的计算资源情况,在实际部署时可能因评估不准确而造成部分客户端的剪枝比例设置欠佳,从而对整体训练质量产生不利影响.

针对 Non-IID 数据挑战,Huang 等人提出的 Fed-STP 算法[7]采用了阶段性训练与剪枝相结合的策略. 该方法将训练过程划分为多个阶段,在不同阶段采用差异化的剪枝强度,从而优化了在非独立同分布数据下的模型性能. 通过这种阶段性设计,Fed-STP 能够更好地适应数据分布差异,提高全局模型的收敛性. 然而,该方法的剪枝决策过程相对复杂,在深层网络上的收敛速度还有提升空间.

近年来,基于数据统计特性的参数剪枝技术逐渐成为研究热点. Yang 等人开发的 DDPruneFL 系统[8]通过优化预训练本地模型来提取关键数据特征,并运用这些特征指导剪枝过程. 该方案不仅优化了计算和通信效率,还充分融合了数据统计特性,有效增强了联邦学习系统对不同数据分布的适应能力. 无论在数据分布均匀或非均匀的情况下,该方法都展现出稳定的性能表现. 但需要指出的是,当面对训练样本较少的客户端时,该技术的泛化性能仍有提升空间,特别是在数据极度匮乏的特殊场景中.

这些剪枝方法各有优势,但在实际应用中仍面临一些共性挑战. 首先是如何在动态变化的网络环境和异构计算资源下保持剪枝策略的稳定性;其次是在保证模型性能的前提下,如何进一步降低通信和计算开销;最后是如何提升在极端 Non-IID 数据分布下的鲁棒性. 这些问题的解决对于推动联邦学习在资源受限场景下的实际应用具有重要意义.

1.2 知识蒸馏优化方法

知识蒸馏技术作为联邦学习中模型压缩的另一重要途径,主要通过构建轻量级学生模型来替代原始复杂模型,从而显著降低通信和计算开销. 近年来,研究者们针对联邦学习的分布式特性,提出了多种创新的知识蒸馏方法,主要包括以下几类技术路线:

Gad 研究团队开发的 FedSKD 方案[9]创新性地支持客户端采用异构规模的本地模型,通过关键知识参数选择性传输机制实现高效协作. 该技术充分适应边缘设备的计算能力差异,使资源受限设备能够部署精简模型架构. 但需指出,由于需要保持多模型架构间的知识一致性,该方案的计算资源消耗较传统联邦学习方法更高,在低性能终端设备上的应用仍面临挑战.

针对特定应用场景的蒸馏方法也取得了显著进展. Yang 研究组提出的 FedDD 方案[10]针对车联网特殊需求,设计了创新的双重知识蒸馏机制. 该技术通过动态选择簇首节点进行中间知识聚合,配合迭代蒸馏过程,显著降低了通信负担. 实测数据显示,相较于传统 FedAvg 方法,该方案能减少千倍级别的通信量. 然而,多轮蒸馏过程会带来一定的信息损失,造成模型准确率小幅降低,在高精度应用场景中需要仔细考量.

无数据知识蒸馏是近年来的研究热点之一. 陈婧等人提出的 DFP-KD 算法[11]摒弃了传统知识蒸馏对原始训练数据的依赖,转而利用生成对抗网络(Generative Adversarial Networks,GAN)合成数据来驱动蒸馏过程. 该方法结合分步指数移动平均(Exponential Moving Average,EMA)更新策略,显著提升了全局模型的训练稳定性和更新速度. 但需要注意的是,这种方法的性能高度依赖于生成数据的质量,当合成数据与真实数据分布存在显著差异时,可能导致知识迁移效果下降.

特征层面的蒸馏方法也展现出独特优势. Zhang 团队提出的 FDL-HAD 方案[12]创新融合了联邦学习与特征蒸馏技术,采用异构感知策略在维持模型准确性的同时显著提升训练速度. 该技术通过在特征层面实现知识传递,有效规避了大规模参数传输的问题. 但研究发现,在网络环境剧烈变动的情况下,该方案的模型适应能力表现不足,仍需改进以更好地应对网络结构的动态变化.

这些知识蒸馏方法虽然在降低通信开销方面取得了显著成效,但仍面临若干关键挑战. 首先是计算开销问题,多数蒸馏方法需要额外的计算资源来维护师生模型间的知识传递;其次是泛化能力局限,在极端 Non-IID 数据分布下模型性能容易波动;最后是动态适应性不足,难以应对网络环境和参与客户端的快速变化. 这些问题的解决对于知识蒸馏在联邦学习中的广泛应用至关重要.

2　通信高效的自适应联邦剪枝优化算法架构

2.1　先验知识

在联邦学习场景下,深度神经网络的参数量庞大导致客户端需要频繁上传完整模型更新参数,造成巨大的通信开销. 设全局模型参数为 w,其中 d 为参数量. 传统联邦学习中,客户端 i 在第 t 轮需上传完整的本地模型更新 Δw_i^t,其通信开销如式(1)所示:

$$C_{comm} = \sum_{t=1}^{T} \sum_{i=1}^{N} \| \Delta w_i^t \|_0 \tag{1}$$

其中 $\| \cdot \|_0$ 表示参数的非零数量,T 为总训练轮次,N 为客户端数量.

核心问题是如何在保证模型性能的前提下,设计高效的参数传输机制,最小化通信开销 C_{comm}. 使用的主要符号说明见表 1.

2.2　系统框架

本文提出的通信高效的自适应联邦剪枝优化方法(CEAFL)采用模型剪枝和知识蒸馏的双重优化策略,来降低联邦学习中的通信成本,并确保剪枝后的模型能够高效聚合并保持性能稳定,CEAFL 算法的整体流程如图 1 所示,有 2 个类型的实体,客户端和服务器,实体的角色说明和系统运行逻辑如下:

表 1　本文主要符号说明

Table 1　Main symbols in this article are explained

符号	说　明
w'^0	初始模型
z_i	当前轮次的参数重要性度量
Z_i	历史参数重要性信息
w_i^t	本地模型
w'^{t+1}	新一轮全局模型
m^t	掩码向量
F_n	损失函数
$g_i(w^t)$	客户端上对于损失函数的梯度
g_j	客户端计算出关于本地模型的梯度
M	参数子集
$\Delta(M)$	所有保留参数的梯度平方之和
$T(M)$	选择参数子集 M 后的总训练时间
c	固定通信延迟
t_j	参数 j 的计算与通信时间开销
$g_i(w_i)$	当前全局模型计算梯度信息
$\odot$	元素级乘积
$\overline{z_t}$	加权平均计算每个客户端的平均重要性
w_s	服务器全局模型参数
x_j	公共数据集中样本
τ	温度系数
$z_s(x_j)$	全局模型在输入 x_j 上的 Logits
$f_k(x_j)$	各个客户端模型最终提取的特征表示
$f_s(x_j)$	全局模型在同一位置的特征表示
h	特征提取器
c_{w_k}	各个客户端模型的分类器
c_{w_s}	全局模型的分类器

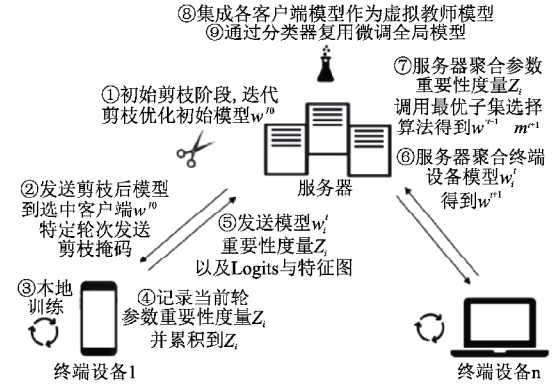

图 1　通信高效的自适应联邦剪枝优化方法流程图

Fig. 1　Flowchart of an adaptive federated pruning optimization method with efficient communication

客户端:客户端集群由多个异构计算节点组成,包括客户端 1 到客户端 n. 这些客户端在本地数据集上进行模型训练,计算并累积参数重要性度量. 它们按照服务器指令上传模型更新、输出和特征图等关键信息,同时接收优化后的全局模型进行下一轮训练.

服务器:服务器是联邦学习系统的核心协调者,负责全局模型的初始化、剪枝优化和聚合更新. 它维护着最优子集选择

算法,执行模型重配置和知识蒸馏等关键操作. 服务器通过周期性接收客户端上传的信息,动态调整模型结构和参数分布.

Step 1. 初始剪枝阶段:服务器启动全局模型的初始化剪枝过程. 通过迭代剪枝算法对初始模型 w'^0 进行优化,基于梯度重要性评估筛选出最具代表性的参数子集.

Step 2. 发送剪枝后模型:服务器将优化后的轻量级模型分发至选定的客户端集群. 在特定训练轮次,服务器会同步发送最新的剪枝掩码,确保所有客户端使用统一的模型结构进行本地训练,保持参数索引的一致性.

Step 3. 本地训练:每个客户端接收到剪枝模型后,基于自身本地数据进行训练. 训练过程中客户端会保持原始模型的参数索引,确保与全局模型的结构对齐.

Step 4. 记录参数重要性:在本地训练过程中,客户端会实时计算并记录当前轮次的参数重要性度量 z_i,同时累积历史参数重要性信息 Z_i.

Step 5. 上传训练结果:训练周期结束后,客户端将更新后的本地模型 w_i^t、累积的重要性度量 Z_i 以及模型输出的 *Logits* 和特征图打包上传至服务器. 这些信息包含了本地数据分布的关键特征和知识.

Step 6. 模型聚合:服务器接收所有客户端上传的模型,执行加权聚合操作. 通过综合考虑各客户端的样本量和数据质量,生成新一代全局模型 w^{t+1},实现知识的有效融合和共享.

Step 7. 重要性聚合与重配置:在特定的重配置轮次,服务器会聚合各客户端上传的参数重要性度量. 通过全局重要性分析和最优子集选择算法,动态调整模型结构,生成新的剪枝掩码 m'^{t+1} 和优化后的模型 w'^{t+1}.

Step 8. 构建虚拟教师模型:服务器集成各客户端的模型输出,构建虚拟教师模型.

Step 9. 知识蒸馏微调:利用虚拟教师模型,服务器通过分类器复用技术对全局模型进行微调. 这一过程结合 KL 散度损失和特征对齐损失,有效缓解了剪枝带来的性能损失.

Step 10. 模型同步:服务器将最终优化的全局模型和更新后的剪枝掩码同步至所有客户端,完成当前训练轮次. 各客户端接收新模型后,即可开启下一轮训练,形成持续优化的闭环流程.

2.3 阶段式自适应模型剪枝算法

为降低联邦学习中因模型参数量过大导致的通信开销与计算资源消耗过高,本节提出一种阶段式自适应模型剪枝算法,通过动态调整模型结构实现高效联邦学习训练. 该方法首先分组遍历所有客户端,生成轻量化初始剪枝掩码,随后在联邦学习过程中联合多客户端数据进一步优化模型,最终在保证模型精度的前提下显著降低资源需求.

首先介绍参数重要性评估算法. 在深度学习模型训练过程中,剪枝算法的核心目标是选择对模型性能贡献最小的参数进行剪枝,从而减小模型的计算和通信开销. 为了做到这一点,需要对每个模型参数的重要性进行评估. 对于当前第 t 轮模型参数 w^t 和掩码向量 m^t,m_j^t 为 0 表示参数 w_j^t 被剪枝,剪枝后的参数不参与后续计算. 客户端计算本地损失函数的梯度如式(2)所示:

$$g_i(w^t) = \nabla F_n(w^t \odot m^t) \tag{2}$$

其中 $\odot$ 表示元素级乘积,$g_i(w^t)$ 表示客户端 i 上对于损失函数 F_n 的梯度,意味着每个模型参数 w_j^t 会根据对应的掩码值 m_j^t 是否为 0 来决定是否参与计算,若 m_j^t 为 0,则 w_j^t 不参与计算,若 m_j^t 为 1,则 m_j^t 保留原始值并参与计算. g_j 是 $g_i(w^t)$ 中的每个分量,即客户端 i 计算出关于参数 w_j^t 的梯度. 通过梯度平方 g_j^2 衡量参数 j 的重要性,其值越大表明该参数对损失下降的贡献越显著.

剪枝决策优化的目标是选择最优参数子集 $M \subseteq \{1,2,\cdots,d\}$,使得在单位时间内损失减少量最大化. 为了实现这一目标,需要综合考虑每个参数的梯度平方 g_j^2 以及该参数的计算和通信时间开销,定义单步训练的 *loss* 减少量如式(3)所示:

$$\Delta(M) = \sum_{j \in M} g_j^2 \tag{3}$$

其中 $\Delta(M)$ 表示在选择的参数子集 M 中所有保留参数的梯度平方之和. 损失函数的减少量与这些参数的贡献紧密相关,因此增加梯度平方较大的参数,能够显著减少损失函数值,提升模型性能.

而单轮训练时间与剩余参数规模呈线性关系,如式(4)所示:

$$T(M) = c + \sum_{j \in M} t_j \tag{4}$$

其中 $T(M)$ 是选择参数子集 M 后的总训练时间,c 为固定通信延迟,t_j 为参数 j 的计算与通信时间开销,在实际应用中,通信延迟和计算开销是影响训练效率的重要因素,因此需要在剪枝决策时加以考虑.

综上所述,剪枝优化问题的目标是通过选择高性价比的参数子集 M,在单位时间内实现最大的损失减少,则优化问题可表述为式(5):

$$M^* = \operatorname{argmax} \frac{\sum_{j \in M} g_j^2}{c + \sum_{j \in M} t_j} \tag{5}$$

为了高效地实现这一优化目标,可以通过贪心算法逐步地选择参数添加到子集 M 中. 该算法的核心思想是按 g_j^2/t_j 排序并逐步选择,仅当新参数的加入使得目标函数值 $\Gamma = \Delta(M)/T(M)$ 不下降时保留该参数,确保每次添加参数均能提升目标函数值,其核心目标是通过迭代筛选高性价比参数,平衡模型精度与训练效率.

具体而言,最优参数子集选择算法可以表示为算法 1.

算法 1. 最优参数子集选择算法

输入: 参数重要性度量 g_j,时间系数 t_j,参数集合大小 S,必须保留的参数集 P

输出: 最优参数子集 A

/ * 先设置 A 为空集 * /

$A = \emptyset$

/ * 以快速排序方式对 $\frac{g_j^2}{t_j}$ 进行降序排序 * /

$S = sort\left(\frac{g_j^2}{t_j}\right)$

/ * 遍历排序后的参数,选择满足条件的最优参数子集 * /

for j in $range[0,S)$:

/ * 仅当选择的参数时目标函数值不下降时,才加入集合 * /

If $\frac{g_j^2}{t_j} \geqslant \Gamma(A \cup \bar{P})$:

$A = A \cup \{j\}$

else
break;
return A;

接下来分阶段详细阐述其设计与实现原理,分为两个阶段,如图 2 所示.

第 1 阶段称为初始剪枝阶段,初始剪枝阶段的目标是生成

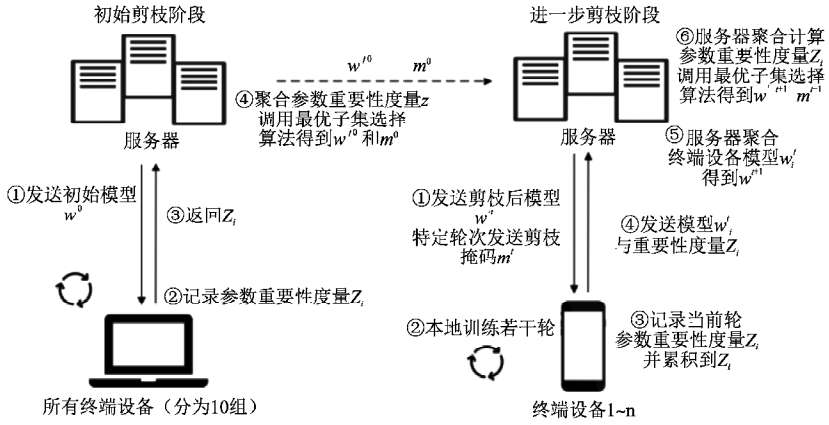

图 2　阶段式自适应剪枝算法流程图

Fig. 2　Flowchart of a phased adaptive pruning algorithm

轻量化初始模型,为后续联邦学习提供高效的起点并降低初期的通信数据量和计算开销. 在这一阶段中,采用不放回采样策略进行迭代剪枝,确保剪枝过程在全局精度最优和计算开销之间取得平衡. 具体而言,循环执行 10 轮剪枝迭代,每轮剪枝遍历不同的客户端,每个客户端 i 基于当前全局模型计算梯度信息 $g_i(w_i)$,服务器汇总梯度信息,评估各参数的重要性度量并更新剪枝模型. 参数重要性评估的方式如式(6)所示:

$$z_i = g_i(w_i) \odot g_i(w_i) \tag{6}$$

其中 g_i 是每个参数梯度,z_i 是客户端 i 的参数重要性度量,反映了参数的重要性. 在此阶段内,仅考虑当前轮次的信息进行剪枝,而不涉及历史重要性度量的累计. 循环执行 10 轮剪枝迭代,逐步减少模型规模,利用最优参数子集算法快速地找到一组在计算效率和模型性能间达到平衡的参数子集. 通过贪心排序与迭代筛选,优先保留梯度贡献大且时间开销小的参数,从而快速压缩模型规模,为后续联邦学习提供高效的初始模型.

在第 2 阶段,也称为进一步剪枝阶段. 在初始剪枝生成的轻量化模型基础上,第 2 阶段通过多客户端协作与周期性重配置进一步优化模型. 此阶段的核心挑战在于如何在数据分布异构且客户端资源受限的条件下,动态调整模型结构以实现全局效率最大化.

具体而言,在每一个本地训练轮次,客户端 i 基于当前全局模型 w^t 和掩码向量 m^t 计算梯度并更新参数,如式(7)所示:

$$g_i(w_i^t) = \nabla F_n(w^t \odot m^t) \tag{7}$$

其中⊙表示元素级乘积,$g_i(w'_i)$ 是客户端 i 对于在当前训练轮次的梯度.

针对低算力设备,客户端根据本地计算能力动态调整训练过程. 首先,通过梯度重要性中位数阈值生成稀疏掩码,如式(8):(top$k$ 表示,k 取前 50%)

$$\tau_{0.5} = \text{top}k(z_i, k = 0.5 \times len(z_i)) \tag{8}$$

其中 $\tau_{0.5}$ 是重要性的中位数阈值,该掩码会保留重要性前 50% 的参数,显著降低计算负载,z_i 是客户端 i 的参数重要性度量,top$k$ 是指选择重要性分数最高的前 50% 参数,$len(\,)$ 为模型参数数量.

然后利用对偶变量来追踪其本地客户端漂移,并通过更新对偶参数来指导本地更新,实现本地目标与全局目标的一致性,如式(9)所示:

$$\lambda_i^{t+1} = \lambda_i^t + \gamma(w_i^{t+1} - w_0^t) \tag{9}$$

其中 λ 为对偶变量,更新值为当前轮次更新的参数,$\lambda = 0.7$ 为低算力设备补偿系数. 对偶变量通过动态追踪本地模型与全局参数的差异,约束客户端更新方向以抑制数据异构性导致的漂移;同时借助历史偏移量的累积补偿,提升联邦学习在非独立同分布数据下的收敛效率与模型稳定性.

为了衡量每个参数对损失函数的贡献,系统会通过梯度平方来计算每个参数的重要性度量,对于中断训练的设备,系统会通过重要性衰减补偿机制维护其贡献. 记录参数重要性度量,如式(10)和式(11)所示:

$$z_i = g_i(w_i'^t) \odot g_i(w_i'^t) \tag{10}$$

$$Z_i = Z_i \cup \{\alpha \cdot z_i\}, \alpha = \min\left(1, \frac{T_{local}}{5}\right) \tag{11}$$

其中 z_i 是客户端 i 的当前轮次参数重要性度量,$g_i(w_i'^t)$ 是客户端的参数梯度,Z_i 是客户端 i 累积的历史参数重要性集合,T_{local} 为完成参与训练轮次,α 为衰减系数,与完成参与训练轮次成正比.

在完成每个客户端的本地训练后,服务器会在聚合轮次对客户端的参数进行加权平均,从而生成全局模型. 具体的全局模型更新过程可以通过式(12)表示:

$$w^{t+1} = \sum_{i=0}^{N-1} p_i w_i^{t+1} \tag{12}$$

当到达周期性重配置轮次时,服务器需要基于各客户端的历史重要性度量来选择最优的参数子集. 首先,服务器收集各客户端 i 的历史重要性度量,并通过加权平均计算每个客户端的平均重要性$\overline{z_i}$,如式(13)所示:

$$\overline{z_i} = \frac{\sum_{z_i \in Z_i} z_i}{|Z_i|} \tag{13}$$

接着,服务器会根据所有参数的贡献,同时利用加权平均方式记录全局重要性度量,如式(14)所示:

$$z = \sum_{i=0}^{N-1} p_i [\, \mathbb{I}_{active}(i) z_i + (1 - \mathbb{I}_{active}(i)) \overline{z_i} \,] \quad (14)$$

其中 p_i 是每个客户端的权重,z 是加权平均后的全局参数重要性度量,$\mathbb{I}_{active}(i)$ 是设备活跃指示函数.

根据全局重要性度量,服务器可以选择最优的参数子集,以提高训练效率并降低通信开销. 选择的过程与算法 1 中描述的最优参数子集选择算法相同.

完成上述步骤后,更新后的模型 w'^{t+1} 将作为下一轮训练的初始模型下发给所有客户端,各客户端基于 w'^{t+1} 进行本地训练,并重复上述过程直至达到预设的轮数 K 或全局模型收敛.

总的来说,阶段式自适应模型剪枝算法如算法 2 所述.

算法 2. 阶段式自适应模型剪枝算法

输入:初始全局模型参数向量 w'^0,初始掩码向量 m^0,联邦学习总训练轮次 K,学习率 η,全局聚合间隔 I,重配置间隔 J,客户端总数 N,选中训练的客户端数 n,客户端掉线率 $p \in [0,1]$

输出:最终全局模型参数 w^K,训练过程中每一轮参数 w^t,掩码 m^t,有效参与率 EPR

$\lambda_i^0 = 0, \forall i \in [0, N-1]$

perm←random_permutation([0,1,…,$N-1$])# 生成随机排列

group_size←[$N/10$]

groups←[perm[i:i + group_size] for i in range(0, N, group_size)]#划分组别

for group in groups:#重要性评估

  for i in group:

    $z_i = g_i(w_i) \odot g_i(w_i)$

    Send z_i to Server

  end for

for t in range[0,$K-1$]:#进一步剪枝阶段

  $Z_n \leftarrow \emptyset, \forall n$#初始化每个客户端的重要性度量

  S_t←从[0,$N-1$]中不放回随机选取 n 个索引

  $S'_t \leftarrow \{ i \in S_t \mid \text{rand}() > p \}$ #实际有效客户端

  if i. type == "low":

    $\tau_{0.5} = \text{topk}(z_i, k = 0.5 \times \text{len}(z_i))$ #重要性中位数

    $m^t_{local} = \mathbb{I}(z_i > \tau_{0.5})$#生成局部掩码

  else:

    $m^t_{local} = m^t$

  $g_i(w'^t_i) = g_i(w'^t_i \odot m^t_{local})$#对每个被选中的客户端,计算掩码梯度与本地参数更新

  $w_i^{t+1} = w'^t_i - \eta(g_i(w'^t_i) \odot m^t_{local})$

  $\lambda_i^{t+1} = \lambda_i^t + (w_i^{t+1} - w_0^t)$#每个被选中的客户端更新其对偶变量以保持目标一致性

  $z_i = g_i(w'^t_i) \odot g_i(w'^t_i)$#记录参数重要性度量

  $Z_i = Z_i \cup z_i$

  if$(t+1)\%I == 0$:#如果到了全局聚合轮次

    $w^{t+1} = \sum_{i=0}^{N-1} p_i w_i^{t+1}$#每个客户端发送当前参数给服务器

  if$(t+1)\%J == 0$:#如果到了重配置轮次

    $\overline{z_i} = \dfrac{\sum_{z_i \in Z_i} z_i}{|Z_i|}$#客户端发送平均参数重要性度量到服务器

    $z = \sum_{i=0}^{N-1} p_i \overline{z_i}$

  else:

    $w_i^{t+1} = w^{t+1}, \forall n$

  $m'^{t+1} \leftarrow m^t$

end for

EPR←$\dfrac{1}{K}\sum_{t=0}^{K-1}\dfrac{|s'_t|}{n}$

Return w^K, $\{w^t\}$, $\{m^t\}$, EPR

阶段式自适应联邦剪枝算法通过动态调整模型结构,有效降低了联邦学习中因模型参数量过大导致的通信与计算资源消耗,从而在保证模型精度的前提下显著提升了训练效率. 此外,该算法通过初始剪枝和进一步剪枝两个阶段,结合参数重要性评估和最优参数子集选择,实现了在异构客户端和数据分布条件下的高效模型优化. 这种方法不仅减少了每轮训练的通信和计算开销,还通过周期性重配置确保了模型在全局范围内的持续优化,为资源受限的联邦学习提供了一种高效的解决方案.

2.4 集成分类器复用的模型微调算法

在上一节模型剪枝的过程中,虽然通过裁剪冗余参数可以有效降低模型规模和推理开销,但也会引入一定程度的性能退化. 对于深度神经网络而言,采用掩码对权重进行硬性剪枝后,模型性能往往受到严重影响,难以维持与原始模型同等水平的预测性能. 此外,剪枝后的模型在适应不同的数据分布时表现较差,泛化能力不足. 本节针对以上问题,设计了集成分类器复用的模型微调算法,通过在服务器利用公共数据及其他可访问数据,将本地模型的集成构建虚拟教师模型,与全局模型之间进行知识转移,对剪枝后的全局模型进行进一步的修正和优化,从而提升模型的泛化能力和鲁棒性,同时增强全局模型对数据分布的感知能力,更好地适应全局数据分布的多样性和复杂性.

在本研究的场景中,各客户端模型由于训练数据差异而掌握了互补的信息. 考虑到每个客户端看数据分布的视角不同,通过知识蒸馏,能够让服务器看到每个客户端模型在公共数据上的预测分布 Logits,并将这些分布融合后蒸馏给全局模型,能够在一定程度上对齐不同客户端的知识,从而取得更好的泛化效果. 此外,通过分类器复用,服务器可以进一步利用客户端的分类器信息,有效整合不同客户端的局部知识,减少模型间的冲突,进一步提升模型的性能.

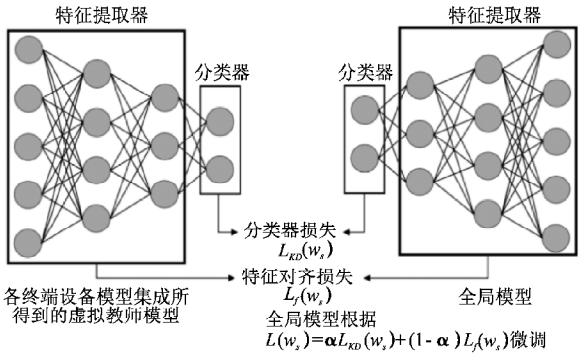

图 3 集成分类器复用的模型微调算法图

Fig. 3 Model fine-tuning algorithm diagram for integrated classifier reuse

具体而言,如图 3 所示,令服务器全局模型参数为 w_s,客户端 i 训练得到的模型参数为 θ_i. 在公共数据集 $D_{pub} = \{(x_j,$

$y_j)\}_{j=1}^{n}$,其中 n 表示公共数据集中的样本数. 服务器全局模型与客户端 i 的模型在 x_j 上的 Logits 输出可表示为式(15)和式(16):

$$z_s(x_j)=f(w_s;x_j) \tag{15}$$

$$z_i(x_j)=f(w_i;x_j) \tag{16}$$

其中 f 表示模型的前向预测参数. 通过将各客户端 Logits 输出平均得到教师分布,虚拟教师模型是所有客户端模型的集成,可表示为式(17):

$$z_t(x_j)=\frac{1}{i}\sum_{i=1}^{i}z_i(x_j) \tag{17}$$

其中 $z_i(x_j)$ 表示客户端 i 对公共数据集中样本 x_j 的 *Logits* 输出. 为方便后续在蒸馏过程中进行概率分布的比较,需要对 Logits 进行 Softmax 归一化. 引入一个温度系数 τ,对虚拟教师模型和全局模型的 Logits 分布进行平滑,避免输出过度尖锐,从而使全局模型在蒸馏过程中更稳定地学习虚拟教师模型的判别信息. 引入温度系数后的虚拟教师分布和全局模型分布可表示为式(18)与式(19):

$$P_t(x_j)=\mathrm{Softmax}\left(\frac{z_t(x_j)}{\tau}\right) \tag{18}$$

$$P_s(x_j)=\mathrm{Softmax}\left(\frac{z_s(x_j)}{\tau}\right) \tag{19}$$

其中 $z_s(x_j)$ 表示全局模型在输入 x_j 上的 Logits. 利用上述分布,即可定义分类器蒸馏损失为两者之间的 KL 散度,如式(20)所示:

$$L_{KD}=\sum_{(x_j,y_j)\in D_{pub}}\mathrm{KL}(p_t(x_j),p_s(x_j)) \tag{20}$$

同时,在模型蒸馏过程中,仅对齐分类器输出的预测分布可能无法充分保留虚拟教师模型在特征表示层面所蕴含的丰富语义信息. 很多关键的分类线索和结构化语义往往在模型中间层已经得到提炼和编码,如果全局模型仅依据最后一层 Logits 学习,容易忽略虚拟教师模型在中间层显式或隐式捕捉到的特征. 因此,通过特征提取器的中间层对虚拟教师模型与全局模型的特征分布进行对齐,全局模型能更直接地获得对数据的深层理解,实现更好的特征表达能力与对感知数据分布能力,从而弥补仅做分类器蒸馏的不足.

因此,本文还选取了虚拟教师模型与全局模型对应的特征提取器的输出施加特征对齐的约束. 具体而言,令 $f_k(x_j)$ 表示各个客户端模型最终提取的特征表示,$f_s(x_j)$ 表示全局模型在同一位置的特征表示,如式(21)与式(22)所示:

$$f_k(x_j)=h\cdot c_{w_i}(x_j) \tag{21}$$

$$f_s(x_j)=h\cdot c_{w_s}(x_j) \tag{22}$$

其中 h 表示特征提取器,c_{w_i} 表示各个客户端模型的分类器,c_{w_s} 表示全局模型的分类器,$\cdot$ 表示特征提取器和分类器的连接操作. 将各客户端的特征表示平均得到虚拟教师模型的特征表示 $f_t(x_j)$,如式(23)所示;特征对齐损失 L_f 可表示为式(24):

$$f_t(x_j)=\frac{1}{i}\sum_{i=1}^{i}f_i(x_j) \tag{23}$$

$$L_f(w_s)=\sum_{(x_j,y_j)\in D_{pub}}\|f_t(x_j)-f_s(x_j)\|_2^2 \tag{24}$$

其中 $\|\cdot\|_2$ 为 L2 范数,用于约束全局模型在此深层特征空间上逼近教师的表征方式. 相较于仅在分类器上进行 Logits 蒸馏,能帮助全局模型更直接地学习到虚拟教师模型所捕捉的高阶语义信息,从而在保持紧凑结构的同时,尽可能保留原模型的判别能力.

同时考虑分类器和特征提取器的蒸馏目标,令 L_{KD} 作为分类器蒸馏损失,L_f 作为特征对齐损失,定义综合损失函数如式(25)所示:

$$L(w_s)=\alpha L_{KD}(w_s)+(1-\alpha)L_f(w_s) \tag{25}$$

其中 $\alpha\in[0,1]$ 作为权重超参数,控制分类器蒸馏与特征对齐之间的平衡. 通过在公共数据集上训练,最小化 $L(w_s)$,可使全局模型在最终预测和特征表示两个层面逼近虚拟教师模型.

集成分类器复用的模型微调算法如算法 3 所示.

算法 3. 集成分类器复用的模型微调算法

输入:全局模型参数 w_s,各个客户端的参数 w_i,公共数据集 D_{pub},蒸馏温度 τ,平衡超参数 α,学习率 η_g

输出:经过微调的全局模型参数 w_s'

for each(x_j,y_j) in D_{pub}:

/* 计算服务器全局模型与客户端模型在公共数据集上的预测分布 */

$z_s(x_j)=f(w_s;x_j)$

for each client i:

$z_k(x_j)=f(w_i;x_j)$

/* 初始化虚拟教师模型的聚合 Logits */

$z_t(x_j)=\frac{1}{i}\sum_{i=1}^{i}z_k(x_j)$

/* 引入温度参数 */

$p_t(x_j)=\mathrm{Softmax}\left(\frac{z_t(x_j)}{\tau}\right)$

$p_s(x_j)=\mathrm{Softmax}\left(\frac{z_s(x_j)}{\tau}\right)$

/* 计算特征分类器特征表示 */

for each client i:

$f_i(x_j)=h\cdot c_{w_i}(x_j)$

$f_t(x_j)=\frac{1}{i}\sum_{i=1}^{i}h_i(x_j)$

$f_s(x_j)=h\cdot c_{w_s}(x_j)$

/* 计算分类器蒸馏损失 */

$L_{KD}=\sum_{(x_j,y_j)\in D_{pub}}\mathrm{KL}(p_t(x_j),p_s(x_j))$

/* 计算特征对齐损失 */

$L_f(w_s)=\sum_{(x_j,y_j)\in D_{pub}}\|f_t(x_j)-f_s(x_j)\|_2^2$

/* 定义损失函数 */

$L(w_s)=\alpha L_{KD}(w_s)+(1-\alpha)L_f(w_s)$

/* 进行模型微调 */

$w_s'=w_s-\eta_g*\nabla_{w_s}L(w_s)$

return w_s'

在集成分类器复用的模型微调算法中,通过在公共数据集上执行虚拟教师模型的 Logits 融合,构建全局一致的软标签,使得全局模型在学习过程中能够更稳定地对齐虚拟教师模型的决策边界. 此外,在蒸馏过程中引入特征对齐机制,使全局模型能够利用特征提取器的输出对齐虚拟教师模型的深层特征,从而有效补偿剪枝带来的信息损失,确保全局模型在参数缩减的情况下仍能保持良好的表示能力. 该方法能够在资源受限和数据异质性环境下,有效缓解模型剪枝带来的性能下降问题,并增强全局模型在联邦学习中的泛化性.

针对民航医疗、金融等隐私敏感领域公共数据难以获取的问题，可启用统计量匹配蒸馏模块：客户端仅上传本地特征分布的均值与方差，服务器依此生成合成特征替代公共数据，避免原始数据暴露；通过约束合成特征与真实特征的分布一致性，如式(26)所示，即使无公共数据，全局模型仍能学习到客户端共享的知识：

$$L_{syn} = \| \tilde{f}_j - f_s(x_j) \|_2^2 + \gamma \cdot KL(\tilde{p}_j \| p_s(x_j)) \quad (26)$$

其中 $\tilde{f}_j$ 表示第 j 个合成特征，$\tilde{p}_j$ 表示合成特征的软标签，$p_s(x_j)$ 表示全局模型对 x_j 的预测分布.

综上所述，统计量匹配蒸馏通过合成特征与分布一致性约束，在无需公共数据的条件下实现了隐私保护与知识迁移的双重目标. 该方法与 CEAFL 的自适应剪枝机制协同优化，既缓解了隐私敏感场景的数据依赖问题，又保持了模型压缩后的泛化性能

3 实验结果与分析

3.1 实验配置与环境

本节以图像分类任务为例，对 CEAFL 框架进行实验仿真. 为全面验证方法在通信资源受限及数据分布 Non-IID 条件下的有效性，实验选用了 MNIST[13]、EMNIST[14]、CIFAR10[15] 和 CIFAR100[16] 这 4 个公共数据集. 其中，MNIST 和 FMNIST 均为灰度图像数据集，每个类别包含约 6000 张训练图和 1000 张测试图，图像尺寸为 28 × 28；而 CIFAR10 和 CIFAR100 为 RGB 图像数据集，尺寸均为 32 × 32，前者包含 10 个类别，每类 5000 张训练图与 1000 张测试图，后者包含 100 个类别，每类 500 张训练图与 100 张测试图. 为更真实地反映联邦学习中的数据异构性，实验中采用了 Non-IID 设置，采用狄利克雷分布进行数据划分，以模拟不同客户端上数据类别分布的不均衡性. 狄利克雷分布是一种常用于联邦学习数据划分的策略，可以通过参数控制数据在不同客户端上的类别分布情况，使不同客户端持有的数据更接近实际场景中的 Non-IID 情况. 具体而言，设定客户端数量为 100，对 MNIST、EMNIST、CIFAR-10 和 CIFAR-100 数据集采用 Dirichlet(α) 分布进行类别划分. 每个类别的数据按照 Dirichlet($\alpha = 0.6$) 分布，将样本分配给不同的客户端. 使用 $\alpha = 0.6$ 进行划分以确保数据异质性，同时避免极端情况，如某些客户端几乎没有数据.

基线对比算法包括 FedAvg、PruneFL[17]、FedLP[18] 和 FedMef[19]，其中 FedAvg 作为联邦学习领域中最经典的方法，通过对各客户端模型参数进行简单平均来实现全局模型更新，具有实现简单、易于部署的特点. PruneFL 主要采用模型剪枝策略来减少模型参数数量，目标是通过剪除对性能贡献较低的冗余参数，来降低模型的计算复杂度和通信量. FedLP 主要采用逐层剪枝技术，在本地训练和联邦更新过程中逐层进行剪枝，在本实验中采用 FedLP-Hetero，该方法在 FedLP 基础上针对异构客户端进行优化，允许不同客户端根据自身计算能力自适应调整剪枝率，进一步降低计算和通信成本，同时保持模型的联邦学习收敛性能. FedMef 采用动态剪枝和激活剪枝技术，通过预算感知剪枝选择关键参数并减少不必要的激活存储，目标是降低存储占用和计算开销，同时保持剪枝后的模型在联邦学习中的准确性和效率.

评价指标主要包括训练准确率、传输模型大小、有效参与率、收敛稳定性以及以浮点运算次数(Floating Point Operations，FLOPs)为指标的计算资源消耗. 训练精度反映了模型在真实任务中的泛化能力，传输模型大小直接衡量了各轮训练中客户端与服务器之间每一轮传输数据量，能直观展示通信开销，而 FLOPs 则用于评估客户端本地训练所需的计算资源，从而体现出模型在资源受限客户端上的适应性.

实验模拟一个中央服务器与 100 个客户端的联邦学习场景，每个全局轮次中，服务器随机选取 10 个客户端参与本轮训练和模型聚合，整个联邦学习过程共进行 300 轮，同时每 50 轮执行一次周期性参数重配置. 在客户端上，本地训练批量大小为 32、学习率设为 0.01，并执行 5 个本地训练轮次. 该配置既能反映实际资源受限环境下的挑战，又为后续对比分析提供了充分的实验数据.

同时为了模拟动态掉线和计算能力差异的情况，本文采用了动态参与机制，每轮除随机选取 10 个客户端外，额外引入掉线模拟，选中客户端有 15% 概率因网络中断停止响应.

3.2 准确率分析

表 2 展示了不同联邦学习算法在 MNIST、EMNIST、CIFAR10 和 CIFAR100 数据集上的平均分类准确率和收敛曲线比较图.

表 2 对比实验中平均准确率比较表

Table 2 Comparison of average accuracy in comparative experiments

算法	MNIST	EMNIST	CIFAR10	CIFAR100
FedAvg	97.91%	94.53%	77.34%	40.83%
PruneFL	96.83%	93.45%	76.73%	37.07%
FedLP-Hetero	96.75%	93.72%	76.51%	39.72%
FedMef	98.01%	94.57%	78.66%	41.52%
CEAFL	98.22%	94.69%	79.51%	42.31%

可以观察到，在简单数据集 MNIST 和 EMNIST 上，各算法的分类准确率均保持在较高水平，MNIST 数据集上的准确率整体在 96% 以上，EMNIST 数据集也达到了 93% 以上，说明这些数据集的分类任务相对简单，大部分方法都能较好地完成训练任务. 然而，在更复杂的 CIFAR10 和 CIFAR100 数据集上，各算法的性能差异开始变得明显，CIFAR10 的准确率分布在 76% 至 80% 之间，而 CIFAR100 由于类别数增多，任务难度更高，各算法的准确率下降至 37% ~43% 之间，展现出更强的模型能力区分度.

FedAvg 作为经典的联邦学习方法，在 MNIST 和 EMNIST 上分别取得了 97.91% 和 94.53% 的准确率，但在 CIFAR10 和 CIFAR100 上分别下降至 77.34% 和 40.83%，表明其在低复杂度数据集上的表现较好，但在更复杂的数据分布和任务设置下，其全局模型难以充分适应不同客户端的异构数据，导致泛化能力受限.

PruneFL 通过剪枝策略减少了模型参数量，在 MNIST 和 EMNIST 上的准确率略低于 FedAvg 分别是 96.83% 和 93.45%，在 CIFAR10 和 CIFAR100 上的表现更差，为76.73% 和 37.07%，说明静态剪枝策略在保证计算和通信效率的同

时,也对模型的表达能力造成了一定损害,尤其是在高维度复杂数据集上,剪枝导致的信息损失影响了模型的最终性能. FedLP-Hetero 采用了逐层剪枝的方法,在 CIFAR10 和 CIFAR100 上分别取得了 76.51% 和 39.72% 的准确率,相较于 PruneFL,在 CIFAR10 上表现稍有下降,而在 CIFAR100 上有所提升,说明该方法能够在一定程度上缓解剪枝导致的信息损失,但仍然难以充分适应数据异质性,尤其在数据类别较多的情况下,剪枝策略的局限性使其模型表达能力仍受到较大限制.

FedMef 采用了动态剪枝策略,并结合预算感知剪枝和缩放激活剪枝技术,使得剪枝过程能够更加灵活地适应不同数据集的特征. 在 MNIST 和 EMNIST 上取得了 98.01% 和 94.57% 的准确率,稍优于 FedAvg 和 PruneFL,表明该方法在低复杂度数据集上的泛化能力较好. 在 CIFAR10 和 CIFAR100 上,其准确率分别为 78.66% 和 41.52%,相比 PruneFL 和 FedLP-Hetero 均有所提升,说明其剪枝策略在一定程度上减少了不必要的信息损失,提高了剪枝后的模型性能. 然而,该方法仍然受到剪枝策略固有局限性的影响,在 CIFAR100 这样高异质性、高类别数的数据集上,其性能仍然有所下降,说明其剪枝策略需要进一步优化,达到更有效地保持模型的表达能力的目的.

与上述方法相比,CEAFL 在所有数据集上均取得了最优的性能,在 MNIST 和 EMNIST 数据集上的准确率分别达到了 98.22% 和 94.69%,在 CIFAR10 和 CIFAR100 上更是分别取得了 79.51% 和 42.31%,超越了所有基线方法.

这主要得益于 CEAFL 采用的阶段式自适应剪枝策略和分类器复用的知识蒸馏方法. 首先,阶段式剪枝策略分为初始剪枝和进一步剪枝两个阶段,在初始剪枝阶段,服务器选取计算能力较强的客户端执行全局初始剪枝,筛选出低贡献参数,从而快速得到轻量化模型;在进一步剪枝阶段,各客户端在本地训练过程中根据数据特性动态更新剪枝掩码,并在服务器进行周期性重配置,使得剪枝策略能够随着训练过程不断优化,从而在保证剪枝效率的同时,尽可能减少对模型表达能力的损害. 其次,在剪枝后 CEAFL 结合了知识蒸馏策略,通过特征层对齐和软目标蒸馏,使得剪枝后的模型能够更好地继承各个终端模型的知识,提高剪枝模型的泛化能力,弥补剪枝过程中可能丢失的重要信息. 这些优化策略的结合,使得 CEAFL 在减少计算和通信成本的同时,仍然能够保持较高的模型精度,特别是在复杂数据集 CIFAR10 和 CIFAR100 上展现出了明显的优势.

综上所述,FedAvg 在算力充足的情况下表现尚可,但其模型结构固定,无法优化计算效率;PruneFL 和 FedLP-Hetero 采用剪枝策略降低了计算和通信开销,但剪枝方法较为静态,导致模型的泛化能力下降;FedMef 采用动态剪枝策略,优化了一定的信息损失,但在高异质性数据环境下仍然存在局限性. 相比之下,CEAFL 通过阶段式剪枝和知识蒸馏的结合,能够在保证剪枝效率的同时提高模型泛化能力,从而在所有数据集上均取得了最优性能,验证了其在资源受限和数据异质性场景下的优势.

3.3　客户端平均通信量与计算量分析

表 3 统计了进一步剪枝阶段的平均每轮通信量. 初始剪枝阶段作为一次性全局剪枝操作,所有算法均需完成相同规模的参数筛选,其通信量相同,故未纳入本表统计. 不同联邦学习算法在进一步剪枝阶段的通信量优化上存在显著差异.

表 3　进一步剪枝阶段通信量比较表(MB)

Table 3　Comparison table of communication volume in further pruning stages(MB)

算法	MNIST	EMNIST	CIFAR10	CIFAR100
FedAvg	3.39	3.73	4.73	5.31
PruneFL	1.53	1.68	2.13	2.39
FedLP-Hetero	2.17	2.39	3.03	3.40
FedMef	0.75	0.82	1.05	1.18
CEAFL	1.29	1.42	1.80	2.02

从通信量来看,FedAvg 的通信开销最高,该方法未引入任何剪枝或压缩机制,每轮都需要传输完整模型,导致较高的通信负担. PruneFL 由于静态剪枝策略的引入,使得通信量在各数据集上相较于 FedAvg 明显减少,例如在 CIFAR10 和 CIFAR100 数据集上,通信量分别降低至 2.13MB 和 2.39MB,但由于剪枝策略固定,剪枝比例未能根据任务动态调整,仍然存在一定的优化空间. FedLP-Hetero 采用逐层剪枝策略,在 CIFAR10 和 CIFAR100 上的通信量进一步降低至 3.03MB 和 3.40MB,相比 FedAvg 有所改善,但剪枝后各客户端的模型结构存在较大差异,使得服务器在聚合时需要额外处理结构异构性,从而影响了通信优化的效果. FedMef 采用了预算感知剪枝和缩放激活剪枝,其剪枝策略在通信量优化方面表现突出,在所有数据集上均达到了最低水平. 在 CIFAR10 和 CIFAR100 数据集上,FedMef 的通信量仅为 1.05MB 和 1.18MB,相较于 FedAvg 降低了近 80%,相比于 PruneFL 也减少了 50% 以上,这表明 FedMef 能够在极大降低通信量的同时维持一定的模型性能.

CEAFL 在通信量优化方面同样表现良好,在 MNIST、EMNIST、CIFAR10 和 CIFAR100 上的通信量分别为 1.29MB、1.42MB、1.80MB 和 2.02MB,相较于 FedAvg 明显减少,相比 PruneFL 和 FedLP-Hetero 也有进一步的优化,尽管通信量略高于 FedMef,但 CEAFL 通过阶段式剪枝和知识蒸馏策略,在减少通信负担的同时,能够更好地保持模型的泛化能力,提升了模型的精度.

每轮计算量比较表如表 4 所示,不同联邦学习算法在客户端的计算量上也存在显著区别.

表 4　客户端计算量比较表(MFLOPs)

Table 4　Comparison of computing quantities of terminal devices(MFLOPs)

算法	MNIST	EMNIST	CIFAR10	CIFAR100
FedAvg	4.30	4.36	9.87	9.99
PruneFL	2.75	2.79	6.32	6.39
FedLP-Hetero	3.56	3.62	8.20	8.29
FedMef	1.04	1.06	2.40	2.43
CEAFL	1.80	1.83	4.15	4.20

从计算量来看,FedAvg 的计算成本最高,在 MNIST 和 EMNIST 上达到 4.30MFLOPs 以上,在 CIFAR10 和 CIFAR100 数据集上更是达到 9.87MFLOPs 和 9.99MFLOPs,主要由于其

未进行任何剪枝优化,导致客户端需要执行完整模型的前向和后向传播计算. PruneFL 由于静态剪枝策略,使得计算量明显降低,在 CIFAR10 和 CIFAR100 数据集上分别减少至6.32 MFLOPs 和 6.39MFLOPs,表明其剪枝策略能够有效减少计算负担,但在不同数据集上的剪枝比例固定,未能根据任务需求灵活调整计算量. FedLP-Hetero 采用逐层剪枝,使得计算量进一步降低,在 CIFAR10 和 CIFAR100 上分别减少至8.20 MFLOPs 和 8.29MFLOPs,但由于其剪枝策略需要额外计算重要性度量,仍然存在较高的计算成本. FedMef 在计算优化方面表现最为突出,其计算量在所有数据集上均最低,在 CIFAR10 和 CIFAR100 上仅为 2.40MFLOPs 和 2.43MFLOPs,相较于 FedAvg 下降了 75% 以上,显示其剪枝策略能够极大地减少计算负担. 然而,FedMef 过于激进的剪枝可能导致模型表达能力下降,尤其是在高维数据集上,剪枝导致的信息损失影响模型最终性能. CEAFL 在计算优化方面也取得了显著的效果,在 MNIST、EMNIST、CIFAR10 和 CIFAR100 数据集上的计算量分别为 1. 80MFLOPs、1. 83MFLOPs、4. 15MFLOPs 和 4.20 MFLOPs,相较于 FedAvg 明显降低,并且相较于 PruneFL 和 FedLP-Hetero 也有进一步优化,虽然计算量略高于 FedMef,但 CEAFL 在剪枝后通过知识蒸馏策略增强了模型的学习能力,使得剪枝后的模型仍然能保持较好的精度,弥补了剪枝导致的信息损失.

因此,CEAFL 通过全局粗筛加局部精修的剪枝策略,结合分类器复用的知识补偿机制,实现了参数重要性感知与计算负载均衡. 实验表明,该方法在异构联邦场景下具有更强的适用性,尤其在高分辨率、高复杂度的数据集中,其通信与计算效率优势更为显著.

3.4 统计量匹配蒸馏的有效性验证

为验证统计量匹配蒸馏在隐私敏感场景下的有效性,本实验在 CIFAR10 和 HAM10000 医疗数据集上进行测试,采用极端 Non-IID 数据划分(每个客户端仅含 1 ~2 个类别). 对比 3 种方法:CEAFL(依赖公共数据)、CEAFL-Stat(统计量匹配蒸馏)和 FedAvg 基线(依赖公共数据). 表 5 展示了 3 种方法的对比结果:

表 5 统计量匹配蒸馏性能对比

Table 5 Performance comparison of statistical matching distillation

方法	CIFAR10 准确率	HAM10000 准确率	特征距离(W)	通信开销(MB/轮)
CEAFL	79.51%	81.2%	–	1.80
CEAFL-Stat	78.63%	80.1%	0.14	2.00
FedAvg	77.34%	75.8%	–	4.73

实验结果表明,CEAFL-Stat 在 CIFAR10 上达到 78.63% 准确率(较原始 CEAFL 仅下降 0.88%),在 HAM10000 医疗数据上保持 80.1% 准确率,显著优于 FedAvg 的 75.8%. 特征距离指标显示合成特征与真实特征的距离仅为 0.14,证明统计量匹配的有效性. 在通信开销方面,安全聚合使每轮通信量从 1.80MB 增加到 2.00MB,仍远低于 FedAvg 的 4.73MB.

这些结果证明,统计量匹配蒸馏在几乎不损失模型性能的前提下,有效缓解了隐私场景下的公共数据依赖问题. 该方法与 CEAFL 原有的自适应剪枝机制协同工作,可为民航、医疗、金融等敏感领域的联邦学习提供了有效的解决方案.

3.5 动态环境下的鲁棒性验证

为验证 CEAFL 算法在真实边缘计算环境中的适应性,本文基于树莓派 4B(Raspberry Pi 4B)的基准性能构建了实验平台. 以树莓派 4B 的算力(四核 Cortex-A72 @ 1. 5GHz,4GB 内存)作为 100% 基准算力,在 CIFAR-10 数据集上部署了包含 100 个异构设备的测试环境. 实验采用动态化配置:每轮训练随机选取 50 个设备参与,其中每个设备有 15% 的独立掉线概率,使得实际参与设备数平均保持在 42 ~43 个. 在算力配置方面,将设备划分为 3 类:30 个树莓派 4B(100% 基准算力)、50 个树莓派 3B +(约 60% 基准算力)以及 20 个树莓派 Zero 2W(约 30% 基准算力). 通过动态电压频率调整和 cgroups 技术精确控制各设备的 CPU 和内存资源,并在低算力设备上设置 40% 的超时失败概率,模拟了边缘计算中常见的性能差异和动态故障场景. 这一基于实际硬件性能梯度的实验设计,为评估算法在真实边缘环境下的适应性提供了可靠验证. 表 6 对比了 CEAFL 与 FedAvg、FedProx 在动态环境下的表现.

表 6 动态环境下的性能对比

Table 6 Comparison of performance in dynamic environments

方法	准确率	有效参与率	低算力利用率	收敛波动
FedAvg	72.1%	68%	32%	±2.3%
FedProx	74.5%	82%	51%	±1.8%
CEAFL	77.8%	89%	63%	±1.2%

结果显示,CEAFL 在准确率(77. 8%)、有效参与率(89%)和低算力设备利用率(63%)上均优于基线方法,且收敛稳定性最佳(波动 ±1.2%). 这表明 CEAFL 的自适应剪枝机制能有效应对设备异构和网络不稳定性,其周期性重配置策略保障了在动态环境下的可靠训练.

4 总 结

本文围绕联邦学习中的计算与通信开销问题,提出了一种通信高效的自适应联邦剪枝优化方法 CEAFL,以在保证模型性能的同时降低通信和计算成本,提高训练效率和泛化能力. 首先,提出了阶段式自适应模型剪枝算法,在初始剪枝阶段,遍历所有客户端筛选出关键参数,构造轻量化初始模型,减少计算和通信负担. 在进一步剪枝阶段,各客户端根据本地数据进行优化,并定期向服务器汇报参数重要性,服务器基于全局统计信息进行周期性剪枝参数重配置,确保剪枝策略持续优化. 其次,提出集成分类器复用的模型微调算法,通过集成客户端模型构建虚拟教师模型,并复用其分类器,增强全局模型对数据分布的感知能力,更好地适应全局数据分布的多样性和复杂性. 最后实验结果表明,相较于传统联邦学习模型压缩方法,CEAFL 在模型精度上较基线方法提升约 0.5% 以上,通信量降低 38% 左右,凸显了该方法在资源受限和数据异质性场景下的综合优势.

References:

[1] McMahan B, Moore E, Ramage D, et al. Communication-efficient

learning of deep networks from decentralized data[C]//Artificial Intelligence and Statistics,2017:1273-1282.

[2] Xu J,Glicksberg B S,Su C,et al. Federated learning for healthcare informatics[J]. Journal of Healthcare Informatics Research,2021,5(1):1-19.

[3] WANG R J,WANG J B,ZHANG F L,et al. Feature map poisoning attack and dual defense mechanism for federated prototype learning[J]. Journal of Software,2024,36(3):1355-1374.

[4] Islam M M,Alawad M. Efficient federated learning through distributed model pruning[C]// IEEE Computer Society Annual Symposium on VLSI,2024:155-160.

[5] CHEN X,QIU H B,LI Y L. Edge-assisted adaptive sparse federated learning optimization algorithm[J]. Journal of Electronics & Information Technology,2025,47(3):1-12.

[6] Wang D W,Hsieh C K,Chan K L,et al. Model pruning for wireless federated learning with heterogeneous channels and devices[C]// VTS Asia Pacific Wireless Communications Symposium,2023:1-5.

[7] Huang Y,Chen W,Zhu S,et al. Fed-STP:an improved federated learning approach via stage-wise training and pruning[C]//2nd International Conference on Cloud Computing,Big Data Application and Software Engineering,2023:108-113.

[8] Yang Y. Towards superior pruning performance in federated learning with discriminative data[J]. IEICE Transactions on Information and Systems,2025,108(1):23-36.

[9] Gad G,Fadlullah Z M,Fouda M M,et al. Federated learning with selective knowledge distillation over bandwidth-constrained wireless networks[C]//International Conference on Communications,2024:3476-3481.

[10] Yang P,Yan M,Cui Y,et al. Communication-efficient federated double distillation in IoV[J]. IEEE Transactions on Cognitive Communications and Networking,2023,9(5):1340-1352.

[11] CHEN J,ZHANG J. Personalized federated learning algorithm based on knowledge distillation without data[J]. Information Network Security,2024,24(10):1562-1569.

[12] Zhang Y,Zhang W,Pu L,et al. To distill or not to distill:toward fast,accurate,and communication-efficient federated distillation learning[J]. IEEE Internet of Things Journal,2023,11(6):10040-10053.

[13] LeCun Y,Bottou L,Bengio Y,et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE,2002,86(11):2278-2324.

[14] Cohen G,Afshar S,Tapson J,et al. EMNIST:extending MNIST to handwritten letters[C]//Interrnational Joint Conference on Neural Networks,2017:2921-2926.

[15] Caldas S,Duddu S M K,Wu P,et al. Leaf:a benchmark for federated settings[J]. arXiv preprint arXiv:1812.01097,2018.

[16] Krizhevsky A,Hinton G. Learning multiple layers of features from tiny images[J]. Handbook of Systemic Autoimmune Diseases,2009,1(4):1-60.

[17] Jiang Y,Wang S,Valls V,et al. Model pruning enables efficient federated learning on edge devices[J]. IEEE Transactions on Neural Networks and Learning Systems,2022,34(12):10374-10386.

[18] Zhu Z,Shi Y,Luo J,et al. Fedlp:Layer-wise pruning mechanism for communication-computation efficient federated learning[C]// IEEE International Conference on Communications,2023:1250-1255.

[19] Huang H,Zhuang W,Chen C,et al. Fedmef:towards memory-efficient federated dynamic pruning[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,2024:27548-27557.

附中文参考文献:

[3] 王瑞锦,王金波,张凤荔,等. 联邦原型学习的特征图中毒攻击和双重防御机制[J]. 软件学报,2024,36(3):1355-1374.

[5] 陈　晓,仇洪冰,李燕龙. 边缘辅助的自适应稀疏联邦学习优化算法[J]. 电子与信息学报,2025,47(3):1-12.

[11] 陈　婧,张　健. 基于知识蒸馏的无数据个性化联邦学习算法[J]. 信息网络安全,2024,24(10):1562-1569.