# 面向大语言模型的高效联邦低秩适应微调方法

张书瑞,陈哲艺,江欣宇

(浙江农林大学 数学与计算机科学学院, 浙江 杭州 311300)

**摘　要:** 大语言模型通过微调可很好地适应特定任务,联邦学习可使用设备的本地数据进行原位计算,进一步增强了隐私保护的微调。现有方法通常采用低秩适应方法进行高效地联邦微调,但依赖传统的联邦学习聚合策略,未充分挖掘大语言模型的潜力,无法很好地解决模型异构配置带来的挑战。为此,提出一种将低秩矩阵乘积差与自适应奇异值分解(SVD)相结合的新方法FedPDS。首先,利用低秩矩阵乘积差引入更多模型参数来扩充表征空间,有效解决了低秩矩阵在下游任务中的表示局限性;其次,在服务器端部署自适应奇异值分解机制,动态重新分配参数以适配异构客户端的计算条件。实验表明,FedPDS在同构、异构联邦环境中显著优于现有的低秩适应方法,收敛速度更快、性能更优。

**关键词:** 联邦学习;大语言模型;低秩适应;模型异构性;奇异值分解

**DOI:** 10. 11907/rjdk. 251823 **扫描二维码阅读全文:**

**中图分类号:** G642 **文献标识码:** A **文章编号:** 1672-7800(XXXX)0XX-0001-07

# Efficient Federated Low-Rank Adaptation Fine-Tuning Method for Large Language Models

ZHANG Shurui, CHEN Zheyi, JIANG Xinyu

(*College of Mathematics and Computer Science*, *Zhejiang A & F University*, *Hangzhou* 311300, *China*)

**Abstract:** Large language models can be effectively adapted to specific tasks through fine-tuning, while federated learning enables in-situ computation using local device data, further enhancing privacy-preserving fine-tuning. Existing methods typically employ low-rank adaptation techniques for efficient federated fine-tuning but rely on traditional federated learning aggregation strategies, failing to fully exploit the potential of large language models and inadequately addressing the challenges posed by heterogeneous model configurations. To this end, a novel approach called FedPDS is proposed, combining the difference of low-rank matrix products with adaptive singular value decomposition (SVD). First, the difference of low-rank matrix products is utilized to introduce additional model parameters, expanding the representation space and effectively overcoming the representational limitations of low-rank matrices in downstream tasks. Second, an adaptive SVD mechanism is deployed on the server side to dynamically reallocate parameters in response to the computational constraints of heterogeneous clients. Experiments demonstrate that FedPDS significantly outperforms existing low-rank adaptation methods in both homogeneous and heterogeneous federated settings, achieving faster convergence and superior performance.

**Key Words:** federated learning; large language model; low-rank adaptation; model heterogeneity; SVD

## 0　引言

在庞大的语料库上进行无监督训练后,语言模型获得了强大的语言能力,即可被定义为大语言模型(large language model, LLM)[1-3]。针对特定任务数据集对大语言模型实施微调,是提升其领域适应性、增强现实场景应用精度的关键环节,但在实验应用中,这些数据分布在多个机构,机构间的数据共享会受到隐私法规和保密问题的限制。联邦学习(federal learning, FL)利用多个机构和分散的数据进行模型训练,无需在外部共享私有数据,为这一挑战提供了解决方案[4]。

**收稿日期:** 2025-12-16

**基金项目:** 国家自然科学基金项目(62306283)

**作者简介:** 张书瑞(1997-),男,浙江农林大学数学与计算机科学学院硕士研究生,CCF学生会员,研究方向为联邦学习和大语言模型;陈哲艺(1991-),男,博士,浙江农林大学数学与计算机科学学院副教授、硕士生导师,研究方向为人工智能和联邦学习;江欣宇(2002-),男,浙江农林大学数学与计算机科学学院硕士研究生,研究方向为联邦学习。本文通讯作者:陈哲艺。

受到本地客户端的计算、存储负担及通信开销等因素限制，在FL框架内微调LLM仍然具有挑战性。为此，部分学者探索了各种参数高效微调技术（Parameter-Efficient Fine-Tuning，PEFT）[5-7]。其中，基于低秩适应（Low-Rank Adaptation，LoRA）的方法因计算效率高、鲁棒性强和出色的灵活性而受到广泛关注[8-11]。

在联邦学习框架下，如何有效聚合LoRA模块中的低秩矩阵仍然挑战巨大。服务器端若直接对客户端上传的低秩矩阵进行平均聚合并广播，将引发聚合误差，进而导致全局模型性能退化。具体而言，在每一轮通信中，客户端仅上传本地更新对应的低秩矩阵，当服务器执行简单平均后，客户端收到的聚合结果与理想全局更新之间必然存在误差，误差随轮次累积，最终对模型收敛精度与稳定性造成影响。此外，由于客户端在数据分布与硬件资源配置上普遍存在异构性，各设备的样本类别、数量及特征空间差异显著，且计算、存储与通信能力参差不齐，这要求LoRA模块必须根据本地系统资源与数据特性，动态调整低秩矩阵的秩。然而，秩的差异化会使各客户端上传的LoRA模块维度互不相同，导致参数服务器难以直接对异构张量执行逐元素聚合，从而限制了直接聚合机制的适用范围。目前，一种直观的方案是对维度较小的矩阵进行零填充，使其与最大秩对齐后再参与聚合，但引入了额外计算与存储开销。例如，在梯度中注入冗余信息，会造成通信轮次延长、收敛速度下降，甚至会引发局部最优陷阱，损害整体学习性能。

本文提出一种支持异构LoRA模块的无聚合误差的联邦微调方法（Federal Learning With Product Difference and SVD，FedPDS）。首先，根据客户端的需要，使用自适应SVD方法产生不同的$B$、$A$矩阵，解决聚合误差和模型异构性；其次，在优化过程中引入更多参数，不仅更新两个低秩矩阵，还更新预训练矩阵，以最大限度激活预训练模型所蕴含的语义知识。预训练矩阵使用两个低阶矩阵乘积的增量，在两次连续迭代中进行更新，有效减轻了仅更新两个低秩矩阵时出现的表征容量不足的问题。更关键的是，所提方法可在优化过程中影响更广泛的预训练权重子空间，充分释放了模型参数潜力，尤其在数据密集型场景中，为复杂分布建模提供了稳定的支持。主要贡献总结为：①研究了在联邦学习环境中使用LoRA微调大语言模型面临的挑战，并对聚合误差进行数学分析；②提出一种基于LoRA的联邦微调算法FedPDS，通过矩阵积差更新预训练模型以增强表征能力，并进一步引入自适应SVD动态重新分配参数，以适配异构客户端的计算条件。

## 1 相关工作

### 1.1 参数高效微调

基础模型在训练和存储方面对设备要求很高，可在预训练模型的基础上引入一组有限的额外可训练参数，以提升其在特定任务上的性能[12]。与适配器方法一样，可训练参数称为预训练模型每一层的适配器，适配器将在微调过程中更新，而主要参数保持不变[13]。还有一种方法只调整偏差，这种方法具有较低的计算成本，适用于计算资源要求极低的场景，或当任务需要对模型进行最小调整时[14]。LoRA是一种PEFT技术，在预训练模型的权重矩阵中引入低秩矩阵实现微调。低秩矩阵具有信息浓缩能力，能使模型在保留原有表达能力的同时快速适配下游任务，适用于需要减少计算资源并保持高性能的场景。本文主要研究LoRA，原因为结构简单、性能优异，实现了与全参数微调相当的性能。

### 1.2 联邦学习中的PEFT

PEFT得益于大幅压缩了客户端与服务器之间通信开销，在FL中得到了广泛应用，核心思想在于将大模型驻留于本地，仅向服务器上传部分关键参数，以极低的带宽实现全局协同的模型训练。例如，Zhang等[15]将基于LoRA的本地更新与FedAvg（联邦平均算法）相结合进行模型聚合。Sun等[16]冻结随机初始化的非零矩阵，只微调零初始化矩阵并更新和聚合一半的参数，以降低通信成本。Qin等[17]通过零阶优化与有限随机种子集，实现了数十亿参数的语言模型全参数微调。本文方法利用异构客户端资源增强全局FL模型的泛化能力，并最大限度提升表征学习能力，以解决当前方法的局限性。

## 2 研究动机

### 2.1 背景知识

在传统的联邦学习框架中，众多客户端共同训练一个全局共享模型，无需共享各自的本地数据，以有效保障数据隐私。具体而言，每个客户端仅利用本地数据进行模型训练，随后将本地模型的更新信息发回中央服务器。服务器则负责收集并聚合来自不同客户端的更新信息，进而迭代优化、完善全局模型。在联邦学习时平均聚合$K$个客户端，从初始模型开始，服务器从各个客户端收集本地更新后的数据，并依据特定算法进行全局更新。

$$\min_{\boldsymbol{w}}\left\{F(\boldsymbol{w})=\sum_{k=1}^{K}p_kF_k(\boldsymbol{w})\right\} \tag{1}$$

式中：$F_k(\boldsymbol{w})$为客户端的目标函数；$p_k$为每个客户端的权重，通常采用算术平均值。

低秩适应方法通过冻结预训练权重并将一对可训练的低秩矩阵注入每个注意力层，以实现大语言模型参数的高效微调[18]。例如，对于预训练的权重矩阵$\boldsymbol{W}$，更新的是一对低秩乘积，只有$\boldsymbol{B}$、$\boldsymbol{A}$是可训练的，且$\boldsymbol{B}$、$\boldsymbol{A}$等于$\boldsymbol{\Delta W}$。在联邦的环境中$\boldsymbol{B}$、$\boldsymbol{A}$由各客户端平均产生，因此该方法不适用于模型异构的场景。综上，在FL中集成LoRA，为协作模型训练提供了一种高效的范式，客户端通过LoRA适配

器，可在资源有限的情况下训练模型，并显著降低通信成本。

$$\boldsymbol{W} = \boldsymbol{W}_0 + \Delta\boldsymbol{W} \tag{2}$$

$$\Delta\boldsymbol{W} = \overline{\boldsymbol{B}}\,\overline{\boldsymbol{A}} = \left(\sum_{k=1}^{K} p_k \boldsymbol{B}_k\right)\left(\sum_{k=1}^{K} p_k \boldsymbol{A}_k\right) \tag{3}$$

### 2.2　误差分析

目前，FedIT、SLoRA 通常将 FedAvg 直接应用于这些低秩矩阵，如图 1 所示[15,19]。第一行为如何使用 LoRA 更新预训练模型；最下面一行为在联邦学习中直接使用 LoRA 的聚合策略。由此可见，该方法并非最优，原因为聚合单个矩阵可能会引入数学误差（见式(4)），将阻碍模型收敛，导致微调成本上升。此外，由于异构数据分布和异构硬件资源，客户端需根据系统和数据异构性调整 LoRA 矩阵的维度，因此该方法也不适合模型异构场景[20]。

$$\frac{1}{2}(\boldsymbol{B}_1\boldsymbol{A}_1 + \boldsymbol{B}_2\boldsymbol{A}_2) \neq \frac{1}{2}(\boldsymbol{B}_1 + \boldsymbol{B}_2)\frac{1}{2}(\boldsymbol{A}_1 + \boldsymbol{A}_2) \tag{4}$$

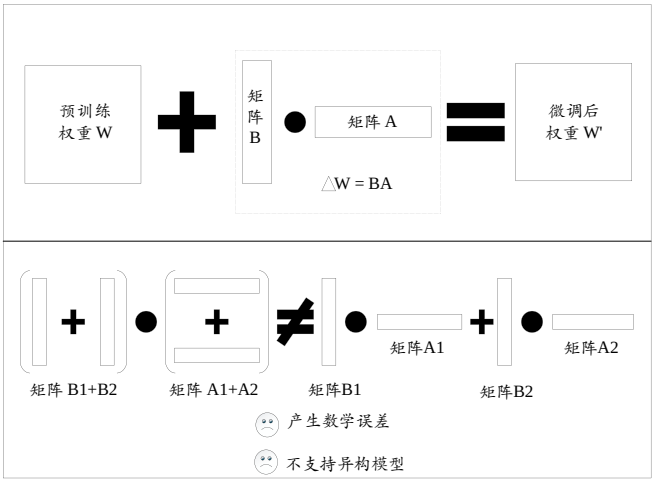



**Fig. 1　Aggregation error generated by the combination of LoRA and FedAvg**

**图 1　LoRA 与 FedAvg 结合产生的聚合误差**

## 3　本文方法

### 3.1　使用低秩矩阵的乘积差更新预训练矩阵

在客户端侧，使用 LoRA 方法优化两个低秩矩阵 $\boldsymbol{B}$、$\boldsymbol{A}$，并获得在反向传播期间生成的梯度。相较于传统 LoRA 方法，本文在更新的同时考虑上一轮状态。

$$\boldsymbol{B}_{t+1} \leftarrow \boldsymbol{B}_t - \eta \cdot g_B - \eta \cdot \boldsymbol{B}_t \tag{5}$$

$$\boldsymbol{A}_{t+1} \leftarrow \boldsymbol{A}_t - \eta \cdot g_A - \eta \cdot \boldsymbol{A}_t \tag{6}$$

式中：$\boldsymbol{B}_t$、$\boldsymbol{A}_t$ 分别为矩阵 $\boldsymbol{B}$、$\boldsymbol{A}$ 在 t 轮的权重；$\eta$ 为学习率；$g_B$、$g_A$ 为矩阵 $\boldsymbol{B}$、$\boldsymbol{A}$ 产生的梯度。

在第 $t$ 轮微调后，每个客户端都会向服务器发回经本地微调后的 LoRA 模块 $\boldsymbol{B}_{t+1}$、$\boldsymbol{A}_{t+1}$。服务器使用这一轮和上一轮的乘积差来更新每个客户端对应的预训练矩阵 $\boldsymbol{W}_t^k$。

$$\boldsymbol{W}_{t+1}^k = \boldsymbol{W}_t^k + \alpha \cdot \left(\boldsymbol{B}_{t+1}^k \boldsymbol{A}_{t+1}^k - \boldsymbol{B}_t^k \boldsymbol{A}_t^k\right) \tag{7}$$

式中：$\alpha$ 为更新比率，一般取 0.5。

该方法将两个矩阵的乘积差作为校正信号，实现了参数空间中预训练矩阵调整方向的精准定位，调整方向不仅与当前任务特有的梯度信息高度契合，还能覆盖更广泛的参数子空间，有效突破了传统低秩微调方法对秩的依赖。同时，该机制重新激活了预训练权重中原本被冻结的通用语义知识，唤醒了模型内部潜在的语义关联与知识结构，实现了对任务相关信息的深层次挖掘与高效利用。

后续，使用算术平均值计算全局预训练权重 $\bar{W}_{t+1}^*$。

$$\bar{W}_{t+1}^* = \sum_{k=1}^{K} p_k W_{t+1}^k,\ \text{s.t.}\ p_k = \frac{1}{K} \tag{8}$$

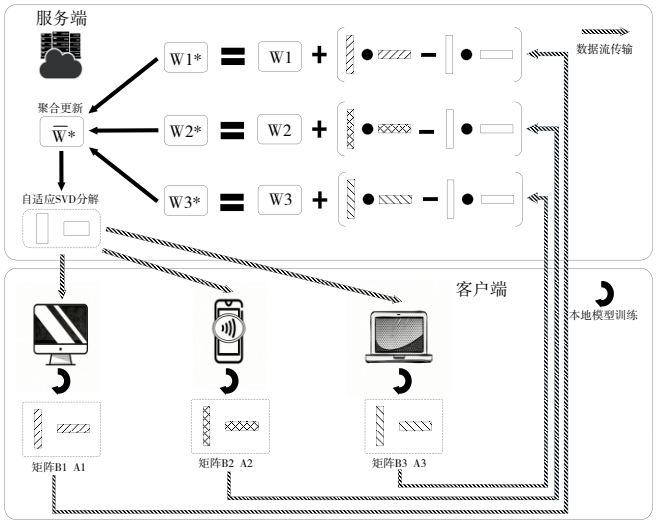



**Fig. 2　FedPDS algorithm framework**

**图 2　FedPDS 算法框架**

### 3.2　自适应 SVD 方法

本文方法相较于传统方法，通过自适应 SVD 动态重新分配参数，以适配异构客户端的计算条件[21]。传统方法受到 LoRA 矩阵秩的限制，参与的客户必须使用相同规模的秩进行聚合，使捕捉异构客户贡献的难度较大。FedPDS 采用一种新颖而直接的方法，根据各客户端的需要对聚合后的矩阵 $\overline{\boldsymbol{W}}_{t+1}^*$ 进行分解。具体而言，将使用 SVD 产生的组件（$U$、$\Sigma$、$V^T$）重新分配给客户端，并根据原始客户端 LoRA 模块的秩尽可能多的保留 $\overline{\boldsymbol{W}}_{t+1}^*$ 的信息。

$$\text{SVD}(\overline{\boldsymbol{W}}_{t+1}^*) = U\Sigma V^T \tag{9}$$

$$\overline{\boldsymbol{W}}_{t+1,k}^* = \underbrace{U[:,:r^k]\,\Sigma[:,r^k,:r^k]}_{\boldsymbol{B}_{t+1}^k}\ \underbrace{V[:,r^k,:]^T}_{\boldsymbol{A}_{t+1}^k} \tag{10}$$

式中：$r^k$ 为每个客户端对应的索引运算符。

接下来，本文使用 top-r 向量组成新的矩阵，各客户端仅需依据本地 LoRA 模块的秩，即可从服务器精准接收对应的聚合知识，并替换原有的低秩矩阵，而本地预训练矩阵在整个过程中始终保持冻结状态，如算法 1 所示。该过程在通信层面开销较小，服务器端仅需下发低秩增量，客户端亦只需更新轻量级适配器。

**算法1** FedPDS算法

输入：预训练参数 $\boldsymbol{W}_0$；初始化低秩矩阵 $\boldsymbol{B}_0, \boldsymbol{A}_0$；学习率 $\eta$；总训练迭代次数 $T$；客户端矩阵秩索引 $r$。

输出：最终全局模型 $\bar{\boldsymbol{W}}_T^*$。

Server Executes：// 服务器执行

for $t$= 0，...，$T$ − 1 do

for each client $k \in \boldsymbol{K}$ in parallel do

$\boldsymbol{B}_{t+1}^k, \boldsymbol{A}_{t+1}^k \leftarrow LocalClientUpdate(\boldsymbol{B}_t^k, \boldsymbol{A}_t^k)$

由式(8)计算出预训练矩阵 $\boldsymbol{W}_{t+1}^k$

end for

$\bar{\boldsymbol{W}}_{t+1}^* \leftarrow Avg(\boldsymbol{W}_{t+1}^k)$ // 加权聚合策略

$\boldsymbol{B}_{t+1}^k, \boldsymbol{A}_{t+1}^k \leftarrow SVD(\bar{\boldsymbol{W}}_{t+1}^*, r^i)$ // 自适应SVD

end for

Client Executes：// 客户端执行

function $LocalClientUpdate(\boldsymbol{B}_t, \boldsymbol{A}_t)$

计算出低秩矩阵梯度 $gB$、$gA$

客户端由式(5)(6)更新低秩矩阵 $\boldsymbol{B}, \boldsymbol{A}$

return $\boldsymbol{B}_{t+1}, \boldsymbol{A}_{t+1}$ // 上传至服务器端

end function

## 4 实验结果与分析

### 4.1 实验设置与基线算法

本文在不同的实验设置和GLUE基准上评估所提方法及其他相关方法在自然语言任务中的性能，需要注意的是：所有客户端的LoRA秩为8、LoRA alpha=16[22,23]。实验在NVIDIA GeForce RTX 4090 GPU上以半精度进行。为了提升模型运行效率，仅展示学习率 $\eta \in \{5E-4, 5E-3, 5E-2\}$ 下的最佳实验结果。

基线模型包括：①FedIT为依赖FedAvg算法进行全局模型聚合，利用本地LoRA模块进行高效微调的模型[15]；②LoRA-FFA为冻结矩阵 $A$，只训练矩阵 $B$ 的模型，牺牲了一半的可训练参数，可显著提升通信效率、计算性能和隐私鲁棒性[16]；③Zero-padding为一种支持异构LoRA的方法，使所有异构的本地LoRA模块扩展到客户端之间的最大的LoRA模块中，并将剩余部分填充为0[24]。

### 4.2 实验结果

#### 4.2.1 自然语言理解任务

为了验证所提方法在分布式环境下的有效性，实验阶段将自然语言理解任务所用数据集随机划分至5个独立客户端，各客户端在本地分别完成模型训练。实验均采用Adam优化器进行模型优化，批大小(batch size)统一设置为32，通信轮次固定为50，以确保实验条件的一致性，实验结果如表1所示。由此可知，FedPDS方法在各项任务中的性能最优，验证了其有效性。

#### 4.2.2 数据异质性

为了系统评估数据异质性对模型性能的影响，基于狄利克雷分布构造3种非独立同分布(non-IID)数据划分方案，并以浓度参数 $d$ 量化异构强度，实验结果如表2、图3所示。其中，$d$=1(记为S-1)、$d$=0.5(记为S-2)、$d$=0.3(记为S-3)，$d$ 越小表明数据异构程度越高。由此可知，FedPDS显著优于基线模型，验证了其在non-IID场景中的鲁棒性。由图3可见，FedPDS在训练的前10轮就已展现出显著优势，随着异构程度加剧，FedPDS的性能优势逐步增大。具体地，在CoLA任务中，FedPDS相较于FedIT准确率依次提升0.90%、1.33%、2.09%；SST-2任务对应的增幅为0.77%、2.10%、2.22%；QNLI任务则分别达到0.29%、1.53%、2.18%。综上，所提方法在数据分布越严重的情况下性能越强。

**Table 1 Performance of different methods on the GLUE benchmark**

**表1 不同方法在GLUE基准上的性能** (%)

| Methods | SST-2 | CoLA | QNLI | QQP | RTE | Average |
|---|---|---|---|---|---|---|
| FedIT | 94.05 | 61.53 | 92.01 | 88.19 | 87.05 | 84.57 |
| FFA-LoRA | 94.89 | 61.54 | 92.87 | 86.98 | 86.43 | 84.54 |
| FedPDS | **95.15** | **62.32** | **93.66** | **89.04** | **88.74** | **85.78** |

**Table 2 Performance of different degrees of data heterogeneity on CoLA、QNLI and SST2 tasks**

**表2 不同程度数据异构性在CoLA、QNLI和SST2任务上的性能(%)**

| Methods | CoLA | | | SST-2 | | | QNLI | | |
|---|---|---|---|---|---|---|---|---|---|
| | S-1 | S-2 | S-3 | S-1 | S-2 | S-3 | S-1 | S-2 | S-3 |
| FedIT | 61.08 | 60.21 | 58.67 | 93.88 | 92.11 | 90.67 | 92.56 | 90.34 | 89.34 |
| FFA-LoRA | 61.04 | 60.48 | 58.94 | 94.01 | 93.31 | 91.34 | 92.44 | 90.76 | 89.02 |
| FedPDS | **61.98** | **61.54** | **60.76** | **94.65** | **94.21** | **92.89** | **92.85** | **91.87** | **91.52** |

#### 4.3.3 模型异构性

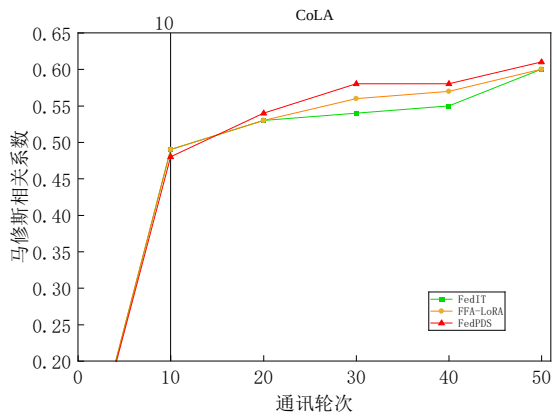
FedPDS相较于FedIT、LoRA-FFA的核心贡献在于：实现了对客户端异构LoRA配置的支持。本文在模型同构设置(FedIT)中，将所有LoRA模块的秩设置为16。在模型异构设置中，将不同的本地LoRA秩(64、32、16、8、4)分别应用于5个客户端，其他设置保持不变，以模拟客户端具有模型异构计算资源的真实场景，如表3所示。由此可知，FedPDS在异构条件下仍能取得与同构几乎一致的精度，验证了其对异构模型的鲁棒性。

值得注意的是，Zero-padding策略在CoLA、SST-2任务会导致性能衰减，准确率分别降低2.54%、2.42%，证明了零填充加剧了在聚合过程中固有的噪声问题，为微调性能造成了损失。

#### 4.3.4 客户数量

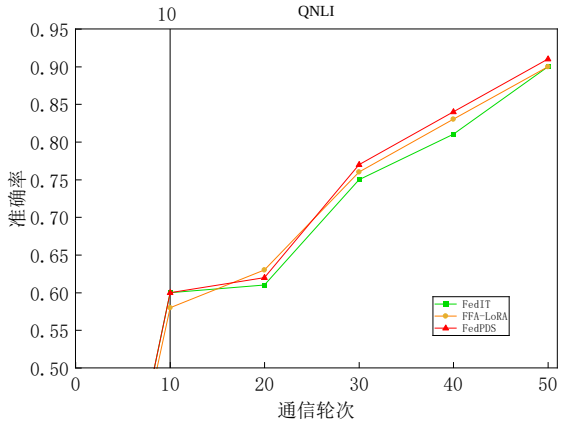
为了验证所提出方法在不同客户端数量下的适应性与鲁棒性，进一步扩展实验至3、5、10个客户端的配置场景，采用与前期实验一致的S-2非IID数据划分方式，并保持其他训练设置不变，实验结果如表4所示。由此可知，在CoLA、QNLI与SST-2等任务上，所提FedPDS方法在不同客户端数量下的性能最优。

在CoLA任务中，FedPDS在3、5、10个客户端设置下的准确率分别为61.69%、61.54%、61.87%；在SST-2、QNLI任务中优势明显。综上，FedPDS不仅在客户端数量较少时



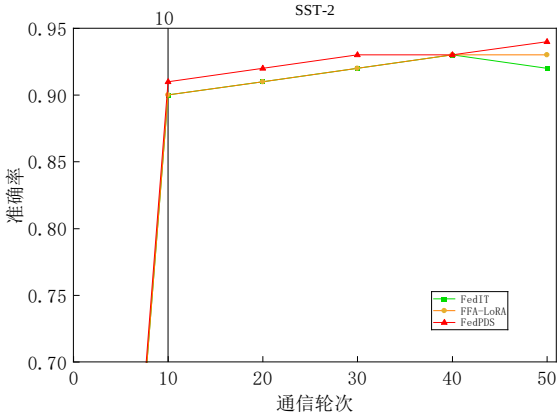

（a） Learning curve of CoLA

（a） CoLA学习曲线



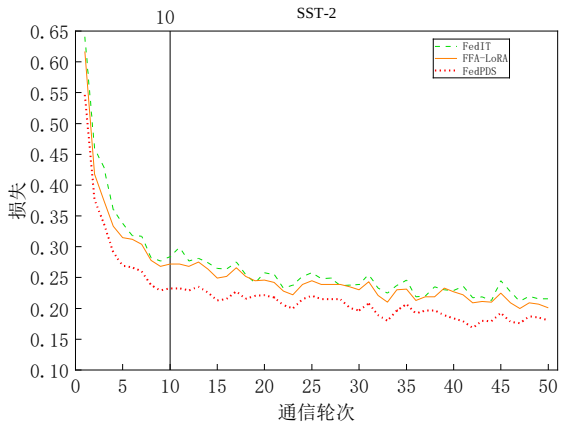

（b） Learning curve of QNLI

（b） QNLI学习曲线




（c） Learning curve of SST-2

（c） SST-2学习曲线




（d） Loss curve of SST-2

（d） SST-2损失曲线

**Fig. 3　Learning curves of CoLA, QNLI, and SST-2 tasks and loss curve of SST-2 task under S-2**

**图3　S-2下CoLA、QNLI和SST-2任务的学习曲线和SST-2任务的损失曲线**

**Table 3　Heterogeneity of models on GLUE benchmark**

**表3　在GLUE基准上各模型的异构性**　　（%）

| Methods | CoLA | SST-2 | QNLI | Average |
|---|---|---|---|---|
| FedIT | 62.87 | 95.26 | 92.46 | 83.53 |
| Zero-padding | 60.33 | 92.84 | 91.12 | 81.43 |
| FedPDS | 62.72 | **95.45** | **92.59** | **83.59** |

表现良好，在客户端数量较多的场景下也能保持稳定的性能优势，证明了所提方法在客户端数量上的适应性和鲁棒性。

**Table 4　Performance on CoLA、QNLI and SST2 tasks with different numbers of clients**

**表4　在CoLA、QNLI和SST2任务上不同客户端数量时的性能（%）**

| Methods | CoLA | | | SST-2 | | | QNLI | | |
|---|---|---|---|---|---|---|---|---|---|
| | K=3 | K=5 | K=10 | K=3 | K=5 | K=10 | K=3 | K=5 | K=10 |
| FedIT | 60.31 | 60.21 | 60.89 | 92.15 | 92.11 | 91.54 | 90.35 | 90.34 | 89.23 |
| FFA-LoRA | 60.67 | 60.48 | 60.05 | 93.41 | 93.31 | 92.43 | 90.86 | 90.76 | 89.43 |
| FedPDS | **61.69** | **61.54** | **61.87** | **94.23** | **94.21** | **93.87** | **90.89** | **91.87** | **89.88** |

### 4.3.5　消融实验

为了检测低秩矩阵乘积差更新预训练矩阵对模型性能的影响，构建FedIT+SVD进行消融研究，如表5所示。由此可知，仅保留SVD的变体，在所有任务上的性能均显著低于FedPDS，表明低秩矩阵乘积差能精细捕捉任务特征，提升模型对不同下游任务的适应能力。

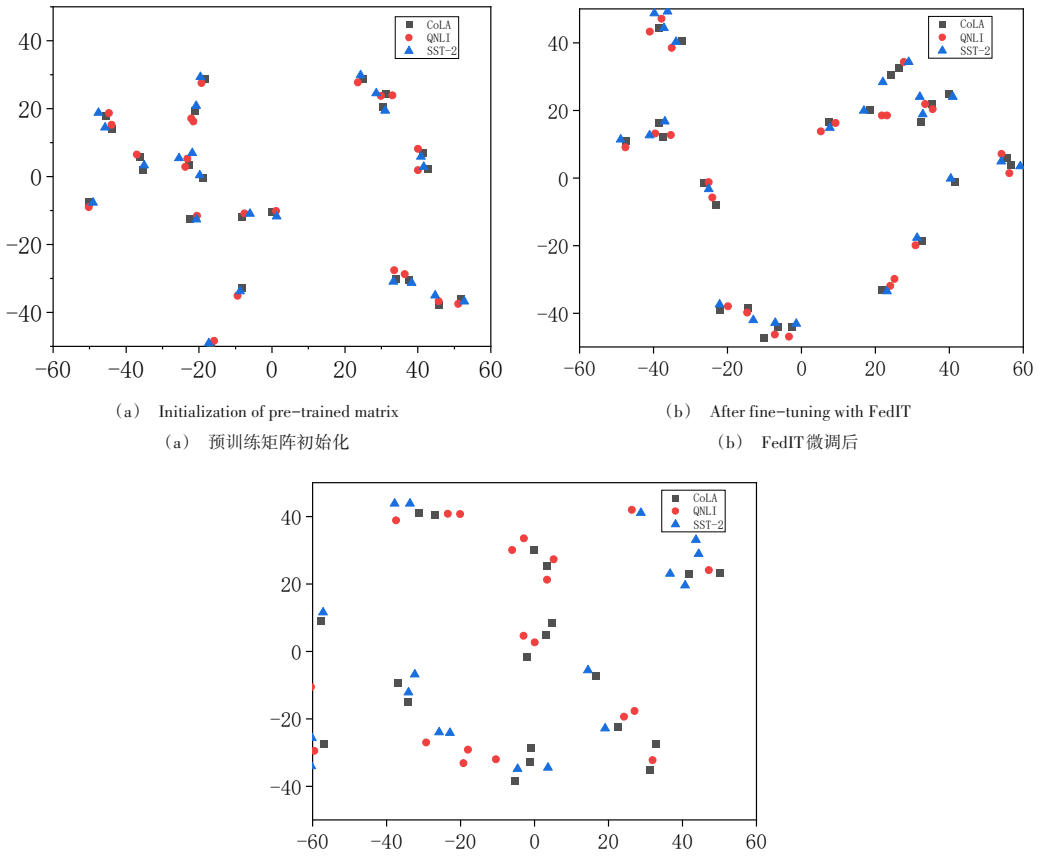
**Table 5　Results of ablation experiment**

**表5　消融实验结果**　　（%）

| Methods | SST-2 | CoLA | QNLI | QQP | RTE | Average |
|---|---|---|---|---|---|---|
| FedIT | 94.05 | 61.53 | 92.01 | 88.19 | 87.05 | 84.57 |
| FedIT+SVD | 94.38 | 61.63 | 92.98 | 88.24 | 87.09 | 84.86 |
| FedPDS | **95.15** | **62.32** | **93.66** | **89.04** | **88.74** | **85.78** |

### 4.3.6　深入分析

为了量化所提方法对预训练参数空间的扰动强度，基于t-SNE可视化策略对GLUE子集（CoLA、SST-2、QNLI）微调前后的预训练权重矩阵进行降维映射，结果如图4所示[25]。由此可见，不同任务共享的预训练权重在初始化阶段呈高度聚集状态，层间差异极小；经FedPDS微调后，参数在空间中的离散度显著高于FedIT，可影响更广阔的参数子空间，从而能更充分地激活预训练模型的内在表征潜能。




（a） Initialization of pre-trained matrix
（a） 预训练矩阵初始化

（b） After fine-tuning with FedIT
（b） FedIT微调后

（c） After fine-tuning with FedPDS
（c） FedPDS微调后

**Fig. 4 T-SNE analysis of per-trained matrix**

**图4 预训练矩阵的t-SNE分析**

## 5 结语

目前，LoRA与联邦学习框架直接结合存在双重挑战：一方面，低秩参数在逐轮平均过程中易放大聚合误差，致使全局模型精度下降；另一方面，相同秩的LoRA模块不适用于模型异构的客户端。为此，本文提出新型联邦微调算法FedPDS。首先，以低秩矩阵乘积差增强模型学习能力，充分挖掘预训练模型潜力；其次，引入自适应SVD动态分配低秩矩阵，实现异构模型的无损聚合。

实验表明，FedPDS在各种场景下均优于现有基线模型。未来，将进一步拓展所提算法的适用范围，探索其在更大规模场景中的应用潜力。

**参考文献：**

[1] MYERS D, MOHAWESH R, CHELLABOINA V I, et al. Foundation and large language models: fundamentals, challenges, opportunities, and social impacts[J]. Cluster Computing, 2024, 27(1): 1-26.

[2] ZHOU Y, RAM P, SALONIDIS T, et al. Single-shot general hyper-parameter optimization for federated learning[DB/OL]. https://arxiv.org/abs/2202.08338.

[3] CHANG Y, WANG X, WANG J, et al. A survey on evaluation of large language models[J]. ACM Transactions on Intelligent Systems and Technology, 2024, 15(3): 1-45.

[4] WEN J, ZHANG Z, LAN Y, et al. A survey on federated learning: challenges and applications[J]. International Journal of Machine Learning and Cybernetics, 2023, 14(2): 513-535.

[5] FU Z, YANG H, SO A M C, et al. On the effectiveness of parameter-efficient fine-tuning[C]// Proceedings of the AAAI Conference on Artificial Intelligence, 2023, 37(11): 12799-12807.

[6] HE H, CAI J, ZHANG J, et al. Sensitivity-aware visual parameter-efficient fine-tuning[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023: 11825-11835.

[7] WANG S W, YU LIN X, LI J. LoRA-GA: low-rank adaptation with gradient approximation[DB/OL]. https://arxiv.org/abs/2407.05000.

[8] LIU S Y, WANG C Y, YIN H, et al. Dora: weight-decomposed low-rank adaptation[DB/OL]. https://arxiv.org/abs/2402.09353.

[9] WANG Z, MA H, ZHAI J. Low-rank adaptation for edge AI[J]. Scientific Reports, 2025, 15(1): 33109.

[10] RAJABZADEH H, VALIPOUR M, ZHU T, et al. Qdylora: quantized dynamic low-rank adaptation for efficient large language model tuning [C]// Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track, 2024: 712-718.

[11] VALIPOUR M, REZAGHOLIZADEH M, KOBYZEV I, et al. DyLo-

RA: parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation [C]// Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, 2023: 3274-3287.

[12] DING N, QIN Y, YANG G, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models [J]. Nature Machine Intelligence, 2023, 5(3): 220-235.

[13] HU Z, WANG L, LAN Y, et al. LLM-adapters: an adapter family for parameter-efficient fine-tuning of large language models [C]// Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 2023: 5254-5276.

[14] CAI H, GAN C, ZHU L, et al. Tinytl: reduce memory, not parameters for efficient on-device learning[J]. Advances in Neural Information Processing Systems, 2020, 33: 11285-11297.

[15] ZHANG J, VAHIDIAN S, KUO M, et al. Towards building the federatedGPT: federated instruction tuning [C]// International Conference on Acoustics, 2024: 6915-6919.

[16] SUN Y, LI Z, LI Y, et al. Improving lora in privacy-preserving federated learning[DB/OL]. https://arxiv. org/abs/2403. 12313.

[17] QIN Z, CHEN D, QIAN B, et al. Federated full-parameter tuning of billion-sized language models with communication cost under 18 kilobytes [DB/OL]. https://arxiv. org/abs/2312. 06353.

[18] HU E J, SHEN Y, WALLIS P, et al. LORA: low-rank adaptation of large language models[DB/OL]. https://arxiv. org/abs/2106. 09685.

[19] BABAKNIYA S, ELKORDY A R, EZZELDIN Y H, et al. Slora: federated parameter efficient fine-tuning of language models [DB/OL]. https://arxiv. org/abs/2308. 06522.

[20] ZHANG R L, DU J H, YIN H. Client selection algorithm in cross-device federated learning[J]. Journal of Software, 2024, 35(12): 5725-5740.
张瑞麟,杜晋华,尹浩. 跨设备联邦学习中的客户端选择算法[J]. 软件学报,2024,35(12):5725-5740.

[21] ABDI H. Singular value decomposition (SVD) and generalized singular value decomposition[EB/OL]. https://personal. utdallas. edu/~herve/Abdi-SVD2007-pretty. pdf.

[22] LIU Y, OTT M, GOYAL N, et al. RoBERTa: a robustly optimized BERT pretraining approach [DB/OL]. https://arxiv. org/abs/1907. 11692.

[23] WANG A, SINGH A, MICHAEL J, et al. GLUE: a multi-task benchmark and analysis platform for natural language understanding[C]// Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 2018: 353-355.

[24] CHO Y J, LIU L, XU Z, et al. Heterogeneous lora for federate-d fine-tuning of on-device foundation models [DB/OL]. https://arxiv. org/abs/2401. 06432.

[25] MAATEN L, HINTON G. Visualizing data using t-SNE[J]. Journal of machine learning research, 2008, 86(9): 2579-2605.

(责任编辑:刘嘉文)