FedProto: 跨异构客户端的联邦原型学习

摘要

在联邦学习（FL）中，客户端之间的异质性通常会在梯度空间中进行客户端知识聚合时阻碍最优化收敛和泛化性能。例如，客户端可能在数据分布、网络延迟、输入/输出空间和/或模型架构方面存在差异，这很容易导致其局部梯度的不对齐。为了提高对异质性的容差，论文提出了一种新颖的联邦原型学习（FedProto）框架，其中客户端和服务器通过抽象的类别原型进行通信，而不是梯度。FedProto 聚合从不同客户端收集的局部原型，然后将全局原型发送回所有客户端，以正则化局部模型的训练。每个客户端的训练旨在最小化局部数据上的分类误差，同时确保生成的局部原型与相应的全局原型足够接近。在实验中，论文提出了一个专为异质性 FL 设计的基准情景，FedProto 在多个数据集上表现良好。此外，复现实验引入了半监督和固定头两种新策略对 FedProto 进行拓展创新，并取得了相应成果。

关键词：联邦学习；数据异质性；原型聚合

1 引言

联邦学习（Federated Learning, FL）是一种分布式机器学习方法，它通过在多个设备上训练模型，而不需要集中存储数据，从而保障了数据隐私和安全性 [30]。在联邦学习（FL）中，当客户端知识在梯度空间中聚集时，客户端间的异构性通常会影响优化的收敛和泛化性能 [16]。例如，客户端可能在数据分布、网络延迟、输入/输出空间和/或模型架构方面存在差异，这很容易导致其局部梯度的不对齐。

尽管已有一些方法尝试解决这些问题，例如通过保持多个全局模型来处理数据异质性 [33]，或通过个性化模型为每个客户端提供专门的模型（个性化联邦学习）[37]，这些方法往往依赖于基于梯度的聚合，这会导致较高的通信成本和对本地模型同质性的强烈依赖。然而，实际应用中，客户端之间的硬件差异使得模型异质性成为普遍现象 [27]，因此这些传统方法存在较大的局限性。

受原型学习的启发，在异构数据集上合并原型可以有效地集成来自不同数据分布的特征表示 [7,23,36]。联邦学习（FL）系统中的客户端智能代理可以通过根据表示交换信息来共享知识，尽管统计和模型存在异质性。例如，当我们谈论“狗”时，不同的人会有一个独特的“想象图片”或“原型”来代表“狗”这个概念。由于不同的生活经历和视觉记忆，他们的原型可能会略有不同。在人与人之间交换这些特定概念的原型可以使他们获得更多关于“狗”概念的知识。将每个 FL 客户端视为类人智能体，本文的方法的核心思想是交换原型而不是共享模型参数或原始数据，这可以自然地匹配人类的知识获取行为。

本文提出的基于原型聚合的方法，不再依赖于模型参数或梯度的聚合，而是通过共享原型来进行信息交换。每个客户端可能有不同的模型架构，但它们可以通过共享代表相同类别的原型来实现有效的知识传递。这种方法能够有效克服由于数据和模型异质性带来的问题，且不需要额外的公共数据集，减少了计算成本。论文的主要贡献包括：

- 提出了一个针对异质性联邦学习的基准设置，考虑了更加广泛的异质性场景。
- 提出了一个创新的基于原型聚合的联邦学习方法，显著提高了在异质性设置下的通信效率。
- 提供了该方法的收敛性保证，并推导了非凸条件下的收敛速度。
- 通过大量实验验证了该方法在通信效率和测试性能上的优势。

该方法为应对各种异质性联邦学习场景提供了一种鲁棒的解决方案，具有较大的应用潜力。

2 相关工作

2.1 异质性联邦学习

客户端之间的统计异质性（也称为非独立同分布问题，non-IID 问题）是联邦学习（FL）面临的最重要挑战之一。FedProx [19] 提出了一个本地正则化项，用于优化每个客户端的本地模型。近期的一些研究 [1, 6, 20] 通过训练个性化模型，利用全局共享信息和个性化部分信息 [15, 37]。第三种解决方案是通过对本地模型进行聚类 [9, 29, 33]，为每个聚类提供多个全局模型。近年来，自监督学习策略被纳入本地训练阶段，以应对异质性挑战 [18, 24, 42]。[8] 则将元训练策略应用于个性化联邦学习。模型架构的异质性是联邦学习中的另一个重大挑战。最近提出的基于知识蒸馏（KD）的联邦学习方法 [14, 17, 21, 26] 为解决这一挑战提供了另一种可选方案。特别地，在假设在联邦设置中增加一个共享的玩具数据集的前提下，这些基于 KD 的联邦学习方法可以从教师模型中蒸馏知识，传递到具有不同模型架构的学生模型中。一些最新的研究还尝试将神经架构搜索与联邦学习相结合 [10, 35, 44]，用于为具有不同硬件能力和配置的客户端群体发现定制的模型架构。[12] 提出了一个集体学习平台，用于处理没有本地训练数据和架构访问的异质性架构。此外，基于功能的神经元匹配方法 [38] 可以聚合具有相似功能的神经元，而不考虑模型架构的差异。然而，以上大多数提到的联邦学习方法仅关注单一的异质性挑战场景。它们都采用基于梯度的聚合方法，这可能引发通信效率低下和梯度攻击等问题 [4, 25, 43, 45]。

2.2 原型学习

原型的概念（多个特征的均值）在多种任务中得到了探索。在图像分类中，原型可以作为一个类别的代理，并通过计算每个类别内特征向量的均值来获得 [36]。在动作识别中，不同时间戳下的视频特征可以被平均，以作为视频的表示 [34, 41]。聚合的本地特征可以作为图像检索的描述符 [2]。将词嵌入平均化作为句子的表示可以在多个自然语言处理基准上取得竞争力的表现 [39]。[13] 中的作者使用原型来表示分布式机器学习中的任务无关信息，并提出了一种新的融合范式来整合这些原型，以生成新任务的模型。在 [31] 中，原型边缘被用于优化

联邦学习中的视觉特征表示。在本文中，我们借用了原型的概念，用来表示一个类别，并在异质性联邦学习的设置中应用原型聚合。总体而言，原型广泛应用于训练样本有限的学习场景 [36]。这一学习场景与跨客户端联邦学习的潜在假设一致：即每个客户端拥有有限的实例，用以独立训练具有所需性能的模型。这一假设得到了基于联邦学习的数据集 [3, 11] 和相关应用，如医疗保健 [32, 40]、街景图像物体检测 [28] 等的广泛支持。

3 本文方法

本文提出了一种基于原型的异质性联邦学习方法，旨在通过交换客户端的原型信息来解决异质性问题。

3.1 本文方法概述

本文提出的解决异构联邦学习的框架的概述如图 1所示。中央服务器接收来自 m 个本地客户的本地原型集 $C_1, C_2, \ldots, C_m$ ，然后通过平均它们来汇总原型。在异构 FL 设置中，这些原型集重叠但不相同。

以 MNIST 数据集为例，第一个客户端识别数字 2、3、4，另一个客户端识别数字 4、5。这是两组不同的手写数字；尽管如此，还是有一个类重叠。服务器自动聚合来自客户端重叠类空间的原型。

在 FL 中使用原型，我们不需要交换梯度或模型参数，这意味着所提出的解决方案可以处理异构模型体系结构。此外，基于原型的 FL 不需要每个客户端提供相同的类，这意味着异构类空间得到了很好的支持。因此，可以解决 FL 中的异质性挑战。

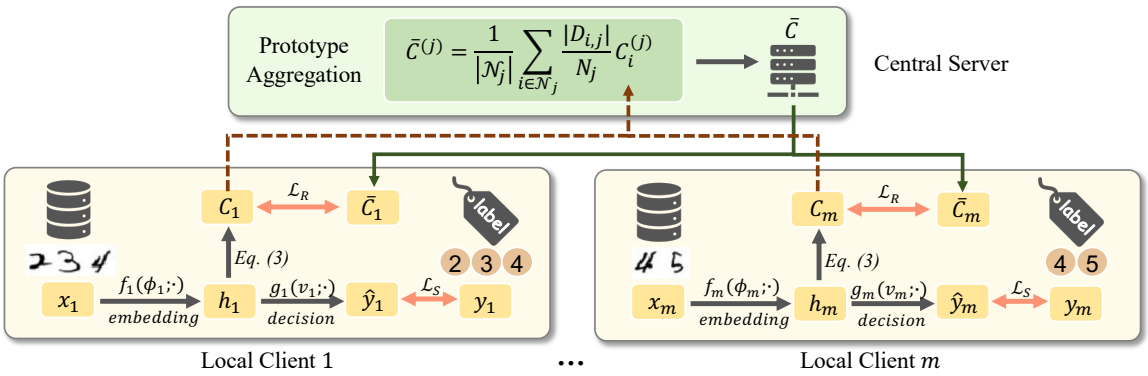

图 1. FedProto 在异构情景中的概览。例如，第一个客户端需要识别数字 $2, 3, 4$，而第 m 个客户端需要识别数字 $4, 5$。首先，客户端通过最小化分类误差损失 $\mathcal{L}_S$ 以及全局原型与局部原型之间的距离 $\mathcal{L}_R$ 来更新其局部原型集。然后，客户端将其原型发送到中央服务器。中央服务器生成全局原型并将其返回给所有客户端，以正则化局部模型的训练。

3.2 基于原型的聚合设置

在原型聚合设置中，异构联邦学习（FL）关注于处理输入/输出空间、数据分布和模型架构的异构性。例如，两个客户端 i 和 k 上的数据集 D_i 和 D_k 可能在标签的统计分布上有所不同。这在安装在移动客户端上的照片分类应用中很常见，服务器需要识别许多类别 $\mathbb{C} =$

$\{C^{(1)}, C^{(2)}, \dots\}$，而每个客户端只需要识别构成 $\mathbb{C}$ 子集的少数类别。尽管客户端之间的类别集可能不同，但它们可能存在重叠。

通常，基于深度学习的模型由两个部分组成：（1）表示层（也称为嵌入函数），用于将输入从原始特征空间转换到嵌入空间；（2）决策层，用于为给定的学习任务做出分类决策。

表示层

第 i 个客户端的嵌入函数是由参数 ϕ_i 参数化的 $f_i(\phi_i)$。记 $h_i = f_i(\phi_i; x)$ 为 x 的嵌入表示。

决策层

给定一个监督学习任务，可以通过由参数 ν_i 参数化的函数 $g_i(\nu_i)$ 生成对 x 的预测。因此，标记函数可以写为 $\mathcal{F}_i(\phi_i, \nu_i) = g_i(\nu_i) \circ f_i(\phi_i)$，用 ω_i 来表示 (ϕ_i, ν_i) 。

原型

定义一个原型 $C^{(j)}$ 来表示第 j 个类。在第 i 个客户端，原型是第 j 类实例的嵌入向量的平均值：

$$C_i^{(j)} = \frac{1}{|D_{i,j}|} \sum_{(x,y) \in D_{i,j}} f_i(\phi_i; x), \tag{1}$$

其中，$D_{i,j}$ 是本地数据集 D_i 的一个子集，由属于第 j 类的训练实例组成。

基于原型的模型推理

在学习任务的推理阶段，我们可以通过测量实例的表示向量 $f(\phi; x)$ 与原型 $C^{(j)}$ 之间的 L2 距离来简单地预测实例 x 的标签 $\hat{y}$：

$$\hat{y} = \arg\min_j ||f(\phi; x) - C^{(j)}||_2. \tag{2}$$

这种原型聚合方法通过在客户端之间共享类别的原型，而不是直接共享模型参数，来有效应对数据和模型的异构性。

3.3 优化目标

FedProto 的优化目标是在分布式网络上解决一个联合优化问题。FedProto 采用基于原型的通信方式，使本地模型可以在与其他本地模型对齐其原型的同时，最小化所有客户端本地学习任务的损失和。该优化目标可以被公式化为：

$$\begin{aligned} &\arg\min_{\{\bar{C}^{(j)}\}_{j=1}^{|\mathbb{C}|}} \sum_{i=1}^{m} \frac{|D_i|}{N} \mathcal{L}_S(\mathcal{F}_i(\omega_i; x), y) + \\ &\lambda \cdot \sum_{j=1}^{|\mathbb{C}|} \sum_{i=1}^{m} \frac{|D_{i,j}|}{N_j} \mathcal{L}_R(\bar{C}_i^{(j)}, C_i^{(j)}), \end{aligned} \tag{3}$$

其中，$\mathcal{L}_S$ 是监督学习的损失，而 $\mathcal{L}_R$ 是一个正则化项，用于测量本地原型 $C^{(j)}$ 和对应的全局原型 $\bar{C}^{(j)}$ 之间的距离（使用 L2 距离）。N 是所有客户端的实例总数，N_j 是所有客户端中属于第 j 类的实例数量。

这个优化问题可以通过交替最小化方法来解决，具体迭代以下两个步骤：(1) 在固定 $\bar{C}_i^{(j)}$ 的情况下，对每个 ω_i 进行最小化；(2) 在固定所有 ω_i 的情况下，对 $\bar{C}_i^{(j)}$ 进行最小化。在分布式设置中，步骤 (1) 简化为每个客户端使用其本地数据进行传统的监督学习，而步骤 (2) 则在服务器端聚合来自本地客户端的局部原型。有关这两个步骤的进一步细节可以在 Algorithm 1 中看到。

Algorithm 1 `FedProto`

Input: $D_i, \omega_i, i = 1, \cdots, m$

服务端执行:

1: 初始化所有类的全局原型集 $\left\{\bar{C}^{(j)}\right\}$.
2: **for** each round $T = 1, 2, ...$ **do**
3: **for** each client i **in parallel do**
4: $C_i \leftarrow$ LocalUpdate $\left(i, \bar{C}_i\right)$
5: **end for**
6: 依照公式. 4 更新全局原型.
7: 使用 $\{\bar{C}^{(j)}\}$ 中的原型更新本地原型集 C_i .
8: **end for**

LocalUpdate$\left(i, \bar{C}_i\right)$:

1: **for** each local epoch **do**
2: **for** batch $(x_i, y_i) \in D_i$ **do**
3: 依照公式 1 计算本地原型.
4: 依照公式 5 使用本地原型计算损失函数.
5: 依照损失函数更新本地模型.
6: **end for**
7: **end for**
8: **return** $C^{(i)}$

3.4 全局原型聚合

在联邦学习中，由于参与客户端的数据和模型的异构性，每个客户端的最优模型参数可能不同。这意味着基于梯度的通信不能为每个客户端提供足够有用的信息。然而，所有客户端共享相同的标签空间，这允许它们共享相同的嵌入空间。通过根据所属类别聚合原型，可以在异构客户端之间高效地交换信息。

对于某个类别 j，服务器从包含该类别的客户端集合中接收原型。在原型聚合操作之后，为类别 j 生成一个全局原型 $\bar{C}^{(j)}$，其计算公式为：

$$\bar{C}^{(j)} = \frac{1}{|\mathcal{N}_j|} \sum_{i \in \mathcal{N}_j} \frac{|D_{i,j}|}{N_j} C_i^{(j)}, \tag{4}$$

其中，$C_i^{(j)}$ 表示来自客户端 i 的类别 j 的原型，$\mathcal{N}_j$ 表示拥有类别 j 的客户端集合。

3.5 本地客户端更新

在本地模型更新过程中，客户端需要更新本地模型以生成跨客户端一致的原型。为此，在本地损失函数中添加了一个正则化项，使得本地原型 $C_i^{(j)}$ 更接近全局原型 $\bar{C}_i^{(j)}$，同时最小化分类误差的损失。损失函数定义为：

$$\mathcal{L}(D_i, \omega_i) = \mathcal{L}_S(\mathcal{F}_i(\omega_i; x), y) + \lambda \cdot \mathcal{L}_R\left(\bar{C}_i, C_i\right), \tag{5}$$

其中，λ 是一个重要性权重，$\mathcal{L}_R$ 是正则化项，定义为：

$$\mathcal{L}_R = \sum_j d(C_i^{(j)}, \bar{C}_i^{(j)}), \tag{6}$$

这里，d 是测量本地生成的原型 $C^{(j)}$ 和全局聚合原型 $\bar{C}^{(j)}$ 之间距离的度量，可以采用多种形式，例如 L1 距离、L2 距离和推土机距离（EMD）等。

4 复现细节

4.1 与已有开源代码对比

本复现工作使用了该论文的开源代码（见 https://github.com/yuetan031/fedproto），并进行了一定的修改，包括：

- 增加了源代码中缺少的基线方法的训练函数。
- 优化了代码结构，将原来集中 main 文件中的数据处理流程及训练函数放到了一个单独的文件中，并添加了一个 MyTrainer 类，使得代码结构更简洁清晰。
- 参考一篇关于联邦半监督方法的论文 [22] 及其开源代码引入了 FedPU 方法的损失函数，并设置了的对应的开关及超参数。
- 参考 FedNH 方法 [5]，引入了对模型网络最后一层线性层参数进行正交初始化并在训练过程中固定的选项。

4.2 实验环境搭建

本实验的代码主要需要以下环境：

- Python 3.6 or greater
- PyTorch 1.6 or greater
- Torchvision
- Numpy 1.18.5
- tensorboard

此外还需要还需要自行下载 mnist、cifar10 数据集并放到相应文件夹下。

4.3 实验运行

在终端使用 bash 命令运行相应程序即可。详见 README.md。

4.4 创新点

本复现工作的创新主要为两个方面：

1. 考虑了现实一些工作中可能不会有太多的标记数据集而存在大量无标签数据，因此本工作进行了 FedProto 方法与半监督方法的结合，并进行了相应实验，尝试利用大量无标签数据进行训练，同时还使用了 FedAvg 方法与半监督方法结合进行对照。

2. 为了进一步对抗数据异质性提升模型预测的准确性并提升个性化及泛化能力，本工作尝试令类别的表示在表示空间中均匀分布，避免因为类别不均衡而导致可能的少数类别的表示与多数类别的表示重叠 [5]，从而对模型网络最后一层线性层参数进行了正交初始化并在客户端训练过程中固定，即使用特定方式初始化的固定头，并进行了相应的实验。

5 实验结果分析

5.1 训练设置

本工作实现了一个典型的联邦学习环境，其中每个客户端拥有自己的本地数据，并通过中央服务器传输/接收信息。使用了两个流行的基准数据集：MNIST 和 CIFAR10。对于 MNIST，使用了一个包含 2 个卷积层和 2 个全连接层的多层卷积神经网络（CNN）；对于 CIFAR10，使用了 ResNet18 模型。

在本地任务中，每个客户端学习一个监督学习任务。为了说明本地任务，借用了小样本学习中的 n-way k-shot 概念，其中 n 控制类别的数量，k 控制每个类别的训练实例数量。为了模拟异构场景，随机改变不同客户端中的 n 和 k 值。定义了 n 和 k 的平均值，然后给每个用户的 n 和 k 添加随机噪声。n 的方差用于控制类空间的异构性，而 k 的方差用于控制数据大小的不平衡。

在联邦学习的基准测试中，研究了 FedProto 在统计和模型异构设置下以及固定头模式的统计异构设置下的性能，并与其他基准方法进行比较，包括 Local（为每个客户端单独训练模型，没有与其他客户端通信）、FedAvg、FedProx、FeSEM 和 FedPer。

在实现细节中，FedProto 和基线方法使用 PyTorch 实现。对于所有数据集，使用了 20 个客户端，每轮通信中所有客户端都会被采样。每个客户端中每个类别的平均大小设定为 100。对于 MNIST 数据集，初始超参数集直接采用了 McMahan 等人的默认超参数集；对于 CIFAR10，使用了在 ImageNet 上预训练的 ResNet18 作为初始模型。预训练网络在 CIFAR10 上的初始平均测试准确性是 27.55%。

5.2 在非独立同分布联邦设置下的表现

在非独立同分布（Non-IID）的联邦学习环境中，FedProto 与其他基准方法进行了比较，这些方法要么是经典的联邦学习方法，要么是特别强调统计异构性的联邦学习方法。所有方法都经过调整以适应这种异构设置。

在统计异构性模拟以及固定头 UH 设置中，假设所有客户端执行的学习任务具有异构的统计分布。为了模拟不同程度的异构性，将标准差固定为 1 或 2，以在类空间和数据大小上创造异构性，这在现实世界的场景中很常见。

在模型异构性模拟中，考虑客户端之间模型架构的细微差异。在 MNIST 和 FEMNIST 数据集中，卷积层的输出通道数设置为 18、20 或 22，而在 CIFAR10 中，不同客户端的卷积层步幅设置不同。这种模型异构性给模型参数平均带来了挑战，因为不同客户端的参数大小不总是相同。

根据图 2的平均测试准确率，本复现结果中 FedProto 在 MNIST 数据集上的训练取得了不错的结果，但在 CIFAR10 数据集上的训练结果的准确度明显比原论文要低。可以看出 FedProto 方法有不错的效果，在一定条件下是最优的方法，同时本工作所改进的固定头的加入使得 CIFAR10 数据集的训练结果的准确度有一定的提升。

Dataset	Method	Stdev of n	Test A verage Acc (×100)			# of Comm Rounds	# of Comm Params (×1000)
			n = 3	n = 4	n = 5		
MNIST	Local	2	98.257±0.400	97.586±0.237	96.773±0.172	0	0
	FeSEM	2	97.551±0.305	97.740±0.825	97.692±0.350	150	430
	FedProx	2	97.743±0.437	**97.488±0.405**	97.746±0.296	110	430
	FedPer	2	98.283±0.378	97.895±0.322	97.735±0.178	100	106
	FedProx	2	97.743±0.437	**97.488±0.405**	97.746±0.296	110	430
	FedAvg	2	98.282±0.309	97.550±0.369	**97.914±0.240**	150	430
	FedProto	2	**98.392±0.306**	97.453±0.357	97.303±0.297	100	4
	FedProto-mh	2	98.316±0.310	97.289±0.433	96.888±0.323	100	4
	FedProto-UH	2	98.114±0.538	97.662±0.221	96.688±0.263	100	4
CIFAR10	Local	1	73.813±2.211	62.593±2.525	60.084±2.270	0	0
	FeSEM	1	53.934±3.911	54.330±2.051	57.422±2.121	150	235000
	FedProx	1	56.408±3.402	54.130±2.609	64.129±1.521	110	235000
	FedPer	1	**77.258±2.924**	**71.850±1.926**	**71.076±2.866**	100	225000
	FedAvg	1	56.555±4.282	53.242±3.933	61.958±1.780	150	235000
	FedProto	1	70.392±2.364	65.084±1.949	59.283±2.328	110	41
	FedProto-mh	1	70.550±2.562	64.528±1.898	59.000±2.605	110	41
	FedProto-UH	1	71.239±2.137	63.712±2.212	59.885±2.780	110	41

图 2. 在两个基准数据集上，采用非独立同分布划分的联邦学习方法对比。最佳结果以粗体显示。可以看出，与基线方法相比，FedProto 在显著减少了通信参数，也实现了不错的准确率。

5.3 可扩展性

FedProto 在不同数量样本情况下的可扩展性表现较好。图 3显示，当客户端可用的样本较少时，测试准确率会持续下降，但 FedProto 的准确率下降速度比 FedAvg 更慢。这表明 FedProto 在不同的数据大小下具有更好的适应性和可扩展性。

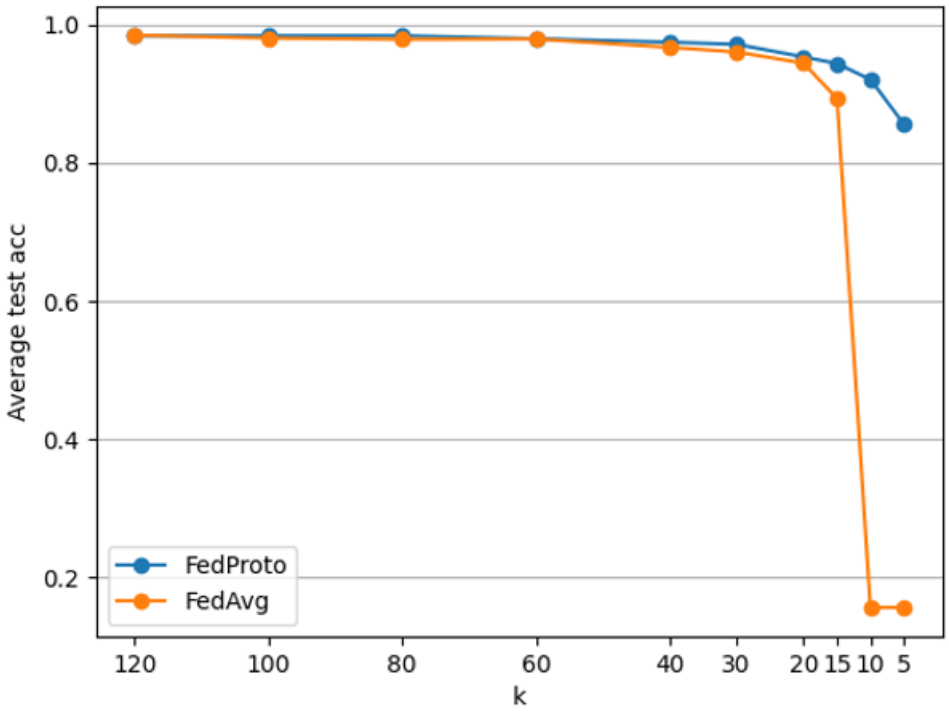

图 3. 在不同类别样本数量变化的情况下，FedProto 和 FedAvg 在 MNIST 数据集上的平均测试准确率。

5.4 半监督设置下的表现

联邦半监督学习（Federated Semi-Supervised Learning, FSSL）是结合了联邦学习和半监督学习两种技术的机器学习方法。在这一框架下，模型在多个设备上进行训练，其中每个设备的数据包括少量的标签数据和大量的未标签数据。

在半监督的模拟中，每个客户端只能在部分类别中标记有限数量的数据。本工作引入了 FedPU 方法 [22] 的损失函数 (puLoss) 来应对标记数据量较少而无标记数据多的半监督学习，结果见图 4。另外，在 CIFAR10 数据集上的训练，本工作还将固定头和半监督方法进行了结合。该实验固定使用 n=3。

可以看出，该半监督方法的设置并不适用于 FedProto 在 MNIST 数据集上的训练，结果准确率会随着 puLoss 权重 pu-weight 的增大而降低，而 FedAvg 却适合使用该 puLoss。而在 CIFAR10 数据集上，当带标签样本占比较小时，可以看出 puLoss 在一定权重的设置下起到了一定作用。

通过固定头 UH 的设置，可以看出 UH 并不适用于 FedAvg 方法，使用后反而会使得准确率降低，但却能够提升 FedProto 方法在 CIFAR10 数据集上训练结果的准确率，但 UH 与 puLoss 结合后却使得该半监督策略对 FedProto 测试结果的准确率产生负面影响。

Dataset	Method	pu_weight	positiveRate				
			1	0.5	0.2	0.1	0.05
MNIST	Fedproto	0	98.392	97.156	94.053	91.433	85.597
		0.5	98.392	96.935	92.808	89.447	79.202
		1	98.392	96.076	90.21	84.488	73.269
	FedAvg	0	98.282	79.328	78.145	74.909	15.705
		0.5	98.282	82.046	79.329	75.19	15.705
		1	98.282	84.769	81.884	76.737	15.705
CIFAR10	FedProto	0	70.392	68.669	49.976	44.765	41.452
		0.1	70.392	68.926	49.437	47.455	41.281
		0.2	70.392	66.435	51.928	46.467	44.445
		0.5	70.392	63.246	57.249	47.494	43.503
		1	70.392	62.43	53.999	47.96	40.826
	FedAvg	0	56.555	53.147	50.418	38.92	37.562
		0.1	56.555	53.325	49.214	39.362	36.187
		0.2	56.555	52.633	47.17	39.723	31.923
		0.5	56.555	55.996	47.357	36.882	36.14
		1	56.555	53.648	44.921	29.937	31.308
	FedProto-UH	0	72.986	68.93	59.03	56.545	51.339
		0.1	72.986	67.883	60.118	50.219	45.22
		0.2	72.986	64.728	58.532	48.797	46.158
		0.5	72.986	60.006	53.291	49.78	45.537
		1	72.986	58.318	50.685	50.016	44.087
	FedAvg-UH	0	54.633	50.745	43.936	38.576	34.623
		0.1	54.633	51.36	44.53	37.211	33.428
		0.2	54.633	53.585	42.801	40.498	35.917
		0.5	54.633	51.023	44.157	36.128	27.361
		1	54.633	51.014	40.941	33.192	27.359

图 4. 联邦半监督设置下实验结果。positiveRate 为带标签数据占比，pu-weight 为 puLoss 权重；pu-weight=0 时相当于仅选用带标签数据进行训练，不使用 puLoss。UH 表示模型使用固定头。

6 总结与展望

论文提出了一种基于原型聚合的新型联邦学习（FL）方法，以应对具有异构输入/输出空间、数据分布和模型架构的挑战性 FL 场景。所提出的方法通过交换原型而非梯度来协同训练智能模型，这为设计基于原型的 FL 提供了新的见解。从理论和实验两个角度全面分析了所提出方法的有效性。

通过复现实验，本工作评估了 FedProto 在联邦学习环境下的性能，特别是在异质性设置下的表现。实验结果表明，FedProto 在 MNIST 数据集上表现优异，能够有效减少通信参数并保持较高的准确率，尤其在非独立同分布环境中，表现显著优于其他基准方法。然而，FedProto 在 CIFAR10 数据集上的表现相对较差，这表明其在复杂任务下仍存在提升空间。另一方面，FedProto 在不同数量样本的情况下展现了较好的可扩展性，当客户端数据减少时，其准确率下降速度相对较慢，证明其在数据规模不均衡时具有较强的适应性。

在使用 puLoss 的半监督学习场景下，FedProto 与 FedAvg 在使用 puLoss 时表现差异明显，尤其是在 MNIST 数据集上，未来可以深入研究如何提升 FedProto 在不同半监督设置下的稳定性和鲁棒性。此外，实验还揭示了 FedProto 在引入了固定头或半监督机制后，可以一定程度优化其在 CIFAR10 数据集标记样本数量较少的情况下的表现，未来可以进一步研究其在复杂任务上应用多种方法的优化机制。

基于这些发现，未来的研究可以着重于优化 FedProto 在 CIFAR10 等复杂数据集上的表现，提升其在半监督学习和模型异构性下的适应性，并探索更为高效的聚合策略，以期在更广泛的异构场景中实现更高的性能。

参考文献

[1] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

[2] Artem Babenko and Victor Lempitsky. Aggregating local deep features for image retrieval. In *Proceedings of the IEEE international conference on computer vision*, pages 1269–1277, 2015.

[3] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečnỳ, H Brendan McMahan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv: 1812.01097*, 2018.

[4] Chen Chen, Jingfeng Zhang, Anthony KH Tung, Mohan Kankanhalli, and Gang Chen. Robust federated recommendation system. *arXiv preprint arXiv:2006.08259*, 2020.

[5] Yutong Dai, Zeyuan Chen, Junnan Li, Shelby Heinecke, Lichao Sun, and Ran Xu. Tackling data heterogeneity in federated learning with class prototypes. *arXiv preprint arXiv:2212.02758*, 2023.

[6] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv:2003.13461*, 2020.

[7] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Selecting relevant features from a universal representation for few-shot classification. 2020.

[8] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. In *Advances in Neural Information Processing Systems*, 2020.

[9] Avishek Ghosh, Jichan Chung, Dong Yin, and Kannan Ramchandran. An efficient framework for clustered federated learning. In *Advances in Neural Information Processing Systems*, 2020.

[10] Chaoyang He, Murali Annavaram, and Salman Avestimehr. FedNAS: Federated deep learning via neural architecture search. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2020.

[11] Chaoyang He, Songze Li, Jinhyun So, Mi Zhang, Hongyi Wang, Xiaoyang Wang, Praneeth Vepakomma, Abhishek Singh, Hang Qiu, Li Shen, et al. Fedml: A research library and benchmark for federated machine learning. *arXiv:2007.13518*, 2020.

[12] Minh Hoang, Nghia Hoang, Bryan Kian Hsiang Low, and Carleton Kingsford. Collective model fusion for multiple black-box experts. In *International Conference on Machine Learning*, pages 2742–2750. PMLR, 2019.

[13] Nghia Hoang, Thanh Lam, Bryan Kian Hsiang Low, and Patrick Jaillet. Learning task-agnostic embedding of multiple black-box experts for multi-task model fusion. In *International Conference on Machine Learning*, pages 4282–4292. PMLR, 2020.

[14] Eunjeong Jeong, Seungeun Oh, Hyesung Kim, Jihong Park, Mehdi Bennis, and Seong-Lyun Kim. Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data. In *Advances in Neural Information Processing Systems*, 2018.

[15] Jing Jiang, Shaoxiong Ji, and Guodong Long. Decentralized knowledge acquisition for mobile internet applications. *World Wide Web*, pages 1–17, 2020.

[16] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, et al. Advances and open problems in federated learning. *arXiv:1912.04977*, 2019.

[17] Daliang Li and Junpu Wang. Fedmd: Heterogenous federated learning via model distillation. In *Advances in Neural Information Processing Systems*, 2020.

[18] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021.

[19] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *MLSys*, 2020.

[20] Paul Pu Liang, Terrance Liu, Liu Ziyin, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *Advances in Neural Information Processing Systems*, 2020.

[21] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. In *Advances in Neural Information Processing Systems*, 2020.

[22] Xinyang Lin, Hanting Chen, Yixing Xu, Chao Xu, Xiaolin Gui, Yiping Deng, and Yunhe Wang. Federated learning with positive and unlabeled data. *INTERNATIONAL CONFERENCE ON MACHINE LEARNING, VOL 162*, 2022.

[23] Lu Liu, William L Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. A universal representation transformer layer for few-shot image classification. In *International Conference on Learning Representations*, 2020.

[24] Yixin Liu, Shirui Pan, Ming Jin, Chuan Zhou, Feng Xia, and Philip S Yu. Graph self-supervised learning: A survey. *arXiv preprint arXiv:2103.00111*, 2021.

[25] Yixin Liu, Shirui Pan, Yu Guang Wang, Fei Xiong, Liang Wang, and Vincent Lee. Anomaly detection in dynamic graphs via transformer. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[26] Guodong Long, Tao Shen, Yue Tan, Leah Gerrard, Allison Clarke, and Jing Jiang. Federated learning for privacy-preserving open innovation future on digital health. In *Humanity Driven AI*. Springer, 2021.

[27] Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. Federated learning for open banking. In *Federated Learning*, pages 240–254. Springer, 2020.

[28] Jiahuan Luo, Xueyang Wu, Yun Luo, Anbu Huang, Yunfeng Huang, Yang Liu, and Qiang Yang. Real-world image datasets for federated learning. *arXiv:1910.11089*, 2019.

[29] Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv:2002.10619*, 2020.

[30] H Brendan McMahan, Eider Moore, Daniel Ramage, et al. Communication-efficient learning of deep networks from decentralized data. *AISTATS*, 2017.

[31] Umberto Michieli and Mete Ozay. Prototype guided federated learning of visual feature representations. *arXiv preprint arXiv:2105.08982*, 2021.

[32] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletari, Holger R Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N Galtier, Bennett A Landman, Klaus Maier-Hein, et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

[33] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 2020.

[34] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014.

[35] Ishika Singh, Haoyi Zhou, Kunlin Yang, Meng Ding, Bill Lin, and Pengtao Xie. Differentially-private federated neural architecture search. In *FL-International Conference on Machine Learning Workshop*, 2020.

[36] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, 30:4077–4087, 2017.

[37] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *arXiv preprint arXiv:2103.00710*, 2021.

[38] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. In *International Conference on Learning Representations*, 2020.

[39] John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. Towards universal paraphrastic sentence embeddings. *arXiv:1511.08198*, 2015.

[40] Jie Xu, Benjamin S Glicksberg, Chang Su, Peter Walker, Jiang Bian, and Fei Wang. Federated learning for healthcare informatics. *Journal of Healthcare Informatics Research*, pages 1–19, 2020.

[41] Guotong Xue, Ming Zhong, Jianxin Li, Jia Chen, Chengshuai Zhai, and Ruochen Kong. Dynamic network embedding survey. *arXiv preprint arXiv:2103.15447*, 2021.

[42] Yaming Yang, Ziyu Guan, Jianxin Li, Wei Zhao, Jiangtao Cui, and Quan Wang. Interpretable and efficient heterogeneous graph convolutional network. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[43] Yu Zheng, Ming Jin, Yixin Liu, Lianhua Chi, Khoa T Phan, and Yi-Ping Phoebe Chen. Generative and contrastive self-supervised learning for graph anomaly detection. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[44] Hangyu Zhu, Haoyu Zhang, and Yaochu Jin. From federated learning to federated neural architecture search: a survey. *Complex & Intelligent Systems*, pages 1–19, 2020.

[45] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In *Advances in Neural Information Processing Systems*, pages 14774–14784, 2019.