DisPFL: Towards Communication-Efficient Personalized Federated Learning via Decentralized Sparse Training

摘要

个性化联邦学习基于解决联邦学习中面临的数据异质性问题以及个性化任务等需求而提出的相应解决方案。本文主要介绍 DisPFL 这一去中心化的个性化联邦学习方法及相关改进尝试。区别于以往联邦学习的客户端-服务端这种中心式通信网络结构，该方法采用去中心化的通信网络结构，并通过个性化稀疏掩码来定制用户的本地模型。这在实现客户端本地模型个性化的同时，由稀疏掩码确定的稀疏模型也有助于实现客户端间通信高效和本地计算的高效性。本次实验相关改进主要基于 DisPFL 中个性化掩码更新与参数聚合策略两个方面进行尝试，并给出后续可能的进一步研究方向。

关键词：个性化联邦学习；稀疏掩码；通信高效；去中心化

1 引言

在有监督学习中，模型的泛化能力依赖于标注数据的规模。虽然我们的世界每天都有海量的数据被创造出来，但它们往往分散在不同的终端或数据中心，且于隐私问题，端与端之间的数据共享是困难，以及海量数据传输成本是高昂的。

联邦学习是近年来兴起的一种分布式机器学习技术，其提出背景是现实生活中数据难以集中管理、隐私安全等突出问题。其核心思想是在多个参与方之间进行模型更新和参数交换，而不需要交换原始数据本身。相较于集中式机器学习中数据需要集中存储在一个中心化服务器上用于模型训练，联邦学习可以在保持机器学习模型的性能和精度同时解决数据隐私和安全问题。

联邦学习算法的研究主要面临以下挑战：

- **通信开销问题**：联邦学习中，各参与方需要与中央服务器进行数据交换，这会导致较大的通信开销。尤其在参与者众多或模型参数较大的情况下，通信开销可能会成为性能瓶颈。
- **隐私保护问题**：联邦学习中，数据在本地进行训练，无需将数据发送至中央服务器，这有助于保护用户隐私。然而，模型更新信息可能泄露参与者的原始数据，因此在训练过程中需要采取额外的隐私保护措施。

- **异质性问题**：包括数据异质、模型异构、系统异构问题。客户端收集的数据可能具有不同的特征分布，这可能导致模型泛化能力下降。同时，由于数据异构性的存在，不同客户端的模型更新可能具有不同的统计特性，这给联邦学习的算法设计带来了挑战。此外，各参与方的用户特征和数据数量格式等可能存在较大差异，这可能导致联邦学习网络中的数据为非独立同分布的。同时，跨设备的数据持有方持有的数据数量可能分布不均匀，这对于数据预处理和模型训练是较大的挑战。在系统上，联邦学习网络中的设备可能存在硬件条件（CPU、内存）、网络连接（3G、4G、5G、WiFi）和电源（电池电量）等方面的差异，这可能导致设备存储、计算和通信能力的不同，从而影响模型的训练效果和效率。

联邦学习的主要研究方向包括：

- **高效通信和计算**：针对联邦学习中的通信开销问题，需要设计高效的通信协议和优化算法，以减少数据交换量并提高训练速度。这包括梯度压缩、模型剪枝、异步更新等技术。
- **隐私保护增强**：隐私保护是联邦学习的核心问题之一, 主要通过引入差分隐私、安全多方计算、同态加密等技术手段，增强联邦学习在数据隐私保护方面的能力。
- **个性化联邦学习**：针对客户端数据非独立同分布问题，该方法通过引入个性化模型、多任务学习、元学习等技术，使全局模型能够更好地适应不同客户端的数据分布，提高模型的泛化能力。
- **激励机制设计**：在联邦学习中，如何激励参与者积极参与训练是一个重要问题。通过设计合理的激励机制，如奖励机制、信誉机制等，鼓励参与者贡献高质量的数据和计算资源。
- **系统优化和部署**：针对联邦学习在实际应用中的部署问题，关注系统优化和可扩展性。这包括资源调度、负载均衡、容错机制等方面的研究，以提高联邦学习系统的稳定性和可靠性。
- **与其他技术的结合**：如物联网、边缘计算、区块链等，以将促进联邦学习在各个领域的应用和发展。

本文主要复现和尝试改进《DisPFL: Towards Communication-Efficient Personalized Federated Learning via Decentralized Sparse Training | ICML 2022》[3] 一文的相关工作。DisPFL一文认为在通常的客户端-服务端这种中心式的网络结构下，当中央服务器出现故障或受到攻击时，整个系统可能会受到影响，并提出一种新的个性化联邦学习框架 DisPFL（Decentralized Personalized Federated Learning），该框架采用去中心化的通信结构，并通过个性化稀疏掩码来定制每个用户的本地模型，在实现客户端本地模型个性化的同时，由稀疏掩码确定的稀疏模型也有助于实现客户端间的通信高效和本地计算的高效。

2 相关工作

2.1 个性化联邦学习

在联邦学习中，所有客户端协同训练一个全局模型，而不共享数据，同时尽量减少通信。但是当客户端的数据分布不同时（Non-IID），学习一个单一的全局模型不能很好地解决问题。用户数据可能具有异质性。在极端情况下，每个客户端可能需要解决不同的任务。为应对客户端异质性数据分布问题，个性化联邦学习（Personalized Federated Learning，PFL）允许每个客户端使用个性化的模型，而不是共享的全局模型。PFL 的关键挑战是如何基于联邦学习进行模型训练，同时允许每个客户端保持自己的独特模型，并限制通信成本。

PFL 的主要目标是为各个客户端提供个性化的全局模型，以解决客户端数据/模型异质性的问题。主要实现方案包括两大类。一类是基于联邦学习获得全局模型，再在本地数据集上做调整（fine-tune）得到个性化模型。该类方案需要考虑如何从数据异质的客户端学习中获得性能较好的全局模型，具体有：（a）从数据集入手，增加数据的同质性，包括采用数据生成器、客户端选择等方案；（b）从模型入手，为本地模型添加正则化项来优化模型学习方向（如 FedProx [10] ，MOON [9] ）、元学习（如 Per-FedAvg [6] ，pFedMe [14] ）、迁移学习（FedMD [8]）。另一类则直接训练本地个性化模型，并修改模型聚合方案用于联邦学习过程中的知识交流，具体有：（a）各客户端使用独立的个性化模型，包括个性化分层网络（LG-FedAvg [11] ）、知识蒸馏（FedMD [8]，FedGen [17] ）等；（b）基于相似性共享模型，如多任务学习、全局模型与本地模型混合策略（APFL [4] ）、基于本地数据分布的分组聚类等。

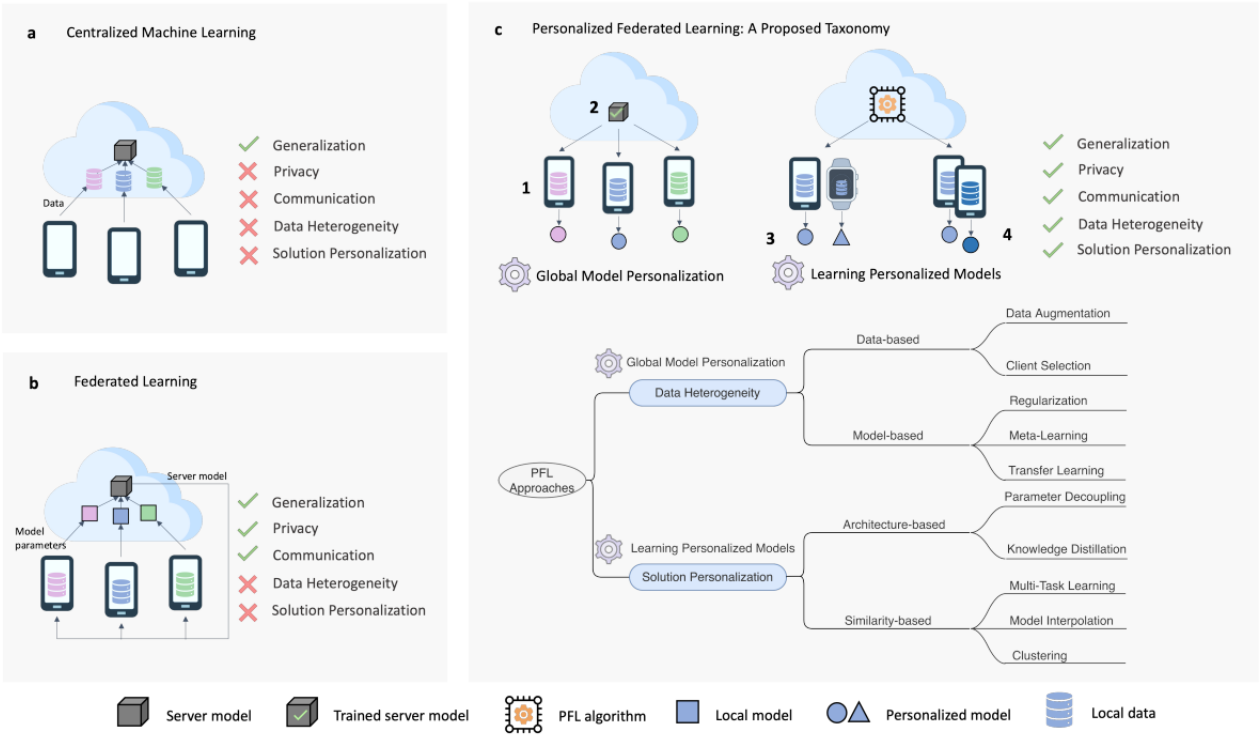

Fig. 1: Concept, Motivations & Proposed Taxonomy for Personalized Federated learning. **a**. Centralized machine learning (CML) which pools data together to train a central ML model. **b**. Federated learning (FL) which trains a global model under the orchestration of a central parameter server. Data resides in different data silos. **c**. Personalized federated learning (PFL) which addresses the limitations of FL through global model personalization and personalized models learning. **1–4** Four categories of PFL approaches: **1**) data-based, **2**) model-based **3**) architecture-based, **4**) similarity-based.

图 1. 联邦学习与个性化联邦学习 [15]

此外，在通信效率上，主要基于量化压缩（Qsgd [1] ）、稀疏化（FedMask [7] ）、以及混合压缩（Qsparse-local-SGD [2] ）等方案来减少通信中传递权重或梯度信息的数据量。

2.2 去中心化学习

去中心化学习采用端到端的通信形式来完成全局模型信息的学习，需要解决数据异质性带来的性能损失问题。此外，需要频繁地通信和聚合客户端更新使其通信效率成为训练的主要瓶颈。在本地训练中进行更多的训练轮次，有助于改进通信效率问题。对于不同的去中心网络结构（如图 2 (b) (c) (d) ），连接越稠密的通信网络有助于得到泛化性更好的模型（ [16] ）。

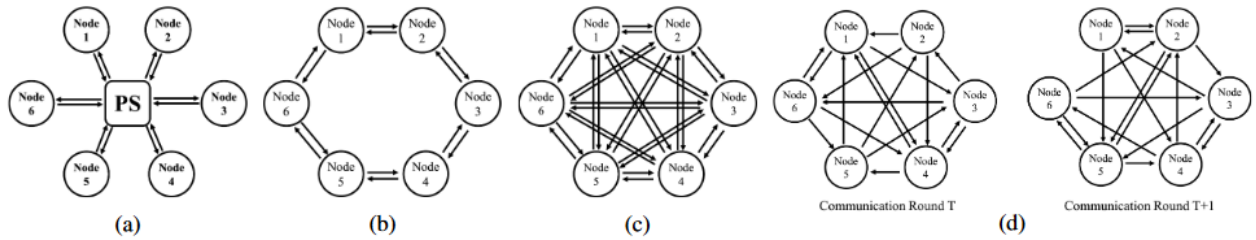

Figure 2. Illustrations of various communication protocols, while (a) represents the centralized parameter server network, (b-d) represents the decentralized setting. (b) denotes the ring topology, (c) denotes the fully-connected topology and (d) denotes the time-varying connected topology.

图 2. 不同的网络通信结构 [3]

2.3 稀疏神经网络

深度神经网络存在过度参数化的特性，一些研究发现使用稀疏模型就可以达到稠密模型的性能。生成稀疏神经网络的方法可分为两大类：从稠密到稀疏（dense-to-sparse）和从稀疏到稀疏（sparse-to-sparse）。从稠密到稀疏的方法基于预训练的稠密模型迭代修剪得到稀疏模型（ [13] ）。而从稀疏到稀疏的方法基于随机初始化的稀疏神经网络，并在训练期间动态调整网络稀疏连接（ [12] ），这类方案也有助于提高训练效率。

3 本文方法

个性化联邦学习的优化目标可表示为以下形式：

$$\begin{aligned} \min_{\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_K\}} f\left(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_K\right) &= \frac{1}{K}\sum_{k=1}^{K} F_k\left(\boldsymbol{w}_k\right), \\ F_k\left(\boldsymbol{w}_k\right) &:= \mathbb{E}\left[\mathcal{L}_{(\boldsymbol{x},y)\sim\mathcal{D}_k}\left(\boldsymbol{w}_k;(\boldsymbol{x},y)\right)\right], \end{aligned} \tag{1}$$

其中，w_k 和 D_k 分别代表第 k 个客户端的个性化模型和数据分布。在实际训练中，通常采用基于本地数据样本的获得的经验风险来评估实际整体风险，该公式可替换为以下形式：

$$\begin{aligned} \min_{\{\boldsymbol{w}_1,\cdots,\boldsymbol{w}_K\}} f\left(\boldsymbol{w}_1,\cdots,\boldsymbol{w}_K\right) &= \frac{1}{K}\sum_{k=1}^{K} \hat{F}_k\left(\boldsymbol{w}_k\right), \\ \hat{F}_k\left(\boldsymbol{w}_k\right) &:= \sum_{i=1}^{n_k} \mathcal{L}\left(\boldsymbol{w}_k;(\boldsymbol{x}_i,y_i)\right), \end{aligned} \tag{2}$$

DisPFL 采用去中心化的通信协议，并基于个性化稀疏掩码来定制每个用户的本地模型。在 DisPFL 中，个性化模型的优化问题可表示为：

$$\min_{\boldsymbol{w},\boldsymbol{m_1},\cdots,\boldsymbol{m_K}} f(\boldsymbol{w},\boldsymbol{m_k}) = \frac{1}{K}\sum_{k=1}^{K} F_k(\boldsymbol{w}\odot\boldsymbol{m_k}),$$
$$F_k(\boldsymbol{w}\odot\boldsymbol{m_k}) := \mathbb{E}\left[\mathcal{L}_{(\boldsymbol{x},y)\sim\mathcal{D}_k}\left(\boldsymbol{w}\odot\boldsymbol{m}_k;(\boldsymbol{x},y)\right)\right], \tag{3}$$

其中，个性化稀疏掩码表示为 $m_k \in \{0,1\}^d$，值为 1 的表示对应权重被激活。

具体来说，DisPFL 首先对每个用户的本地数据进行个性化特征提取，然后根据个性化特征生成对应的个性化稀疏掩码。在本地训练过程中，每个用户只保留掩码中指定的参数进行更新，从而实现了个性化的稀疏训练。在节点间通信过程中，每个用户只传输掩码中指定的参数和对应的梯度信息，从而减少了通信量。具体过程如算法 1（图 4 ）所示。

1. 客户端持有本地模型权重参数和对应的个性化稀疏掩码。其中个性化稀疏掩码，根据给定的模型整体稀疏度，再基于 Erdos-Renyi Kernel (ERK) [5] 方案（参数更多的层使用更高的稀疏度，参数较少的层使用更小的稀疏度）为每层选择一个较为合适的稀疏度后，随机初始化每层稀疏掩码。

2. 客户端将接收到邻节点（根据网络结构决定，这里选择随时间动态变化的（即随机的）网络结构）的模型权重及其相对应的个性化稀疏掩码，依照个性化稀疏掩码计数对权重加权聚合（包括自身的），再使用本地个性化稀疏掩码提取有关权重用于训练。

3. 训练过程中，只使用稀疏掩码所确定的参数用于训练和梯度更新（稀疏模型可减少计算量）。

4. 一轮训练后，通过更新个性化稀疏掩码调整模型参数结构（图 3 ），以期得到更契合本地数据分布的稀疏模型。个性化稀疏掩码的更新主要包括两阶段：(a)Pruning：DisPFL 论文基于余弦退火的方案（公式 4 ，T_{end} 与 t 分别为总/当前通信轮次）给出参数删除比例，将相应比例的最小一部分的权值删除；(b)Re-grow：从未被选择的参数中随机选取，或者使用本地一个 batch 数据训练一次，获得梯度用于决策。

$$f_{decay} = \frac{\alpha}{2}(1 + cos(\frac{t\pi}{T_{end}})) \tag{4}$$

5. 重复过程 2 - 4 一定轮次后使各本地模型收敛。

Algorithm 2 Local mask searching

Input: $\boldsymbol{w}_{k,t+1}$ and corresponding mask $\boldsymbol{m}_{k,t}$
Output: New mask $\boldsymbol{m}_{k,t+1}$
Compute current prune rate α_t using cosine annealing principle with initial pruning rate α_0
Sample a batch of local data and do loss backward to get the dense gradient $g(w_{k,t+1})$
for layer $j \in J$ do **do**
  Update mask $m^j_{k,t+\frac{1}{2}}$ by zeroing out α_t-proportion of weights with magnitude pruning
  Update mask $m^j_{k,t+1}$ via recovering weights with gradient information $g(w_{k,t+1})$
end for
Get new mask $m_{k,t+1}$

图 3. 本地个性掩码更新过程 [3]

Algorithm 1 Dis-PFL

Input: Total number of clients K; Each client's capacity c_k; Total communication rounds T
Initialization: Randomly initialize each client's model $\boldsymbol{w}_{i,0}$ and its mask $\boldsymbol{m}_{i,0}$ according to c_k
Output: Personalized local models $\boldsymbol{w}_{k,T}$
for $t = 0$ to $T-1$ do **do**
  for node k in parallel **do**
    Receive neighbors' models $\boldsymbol{w}_{j,t}$ and corresponding masks $\boldsymbol{m}_{j,t}$ from neighborhood set $S_{k,t}$
    $\boldsymbol{w}_{k,t+\frac{1}{2}} = \left(\frac{\boldsymbol{w}_{k,t} + \sum_{j \in S_{k,t}} \boldsymbol{w}_{j,t}}{\boldsymbol{m}_{k,t} + \sum_{j \in S_{k,t}} \boldsymbol{m}_{j,t}} \right) \odot \boldsymbol{m}_{k,t}$
    $\widetilde{\boldsymbol{w}}_{k,t,0} = \boldsymbol{w}_{k,t+\frac{1}{2}}$
    for $\tau = 0$ to $N-1$ do **do**
      Sample a batch of data $\xi_{k,t,\tau}$ from local dataset
      $g_{k,t,\tau}(\widetilde{\boldsymbol{w}}_{k,t,\tau}) = \nabla_{\widetilde{\boldsymbol{w}}_{k,t,\tau}} L(\widetilde{\boldsymbol{w}}_{k,t,\tau}; \xi_{k,t,\tau})$
      $\widetilde{\boldsymbol{w}}_{k,t,\tau+1} = \widetilde{\boldsymbol{w}}_{k,t,\tau} - \eta \boldsymbol{m}_{k,t} \odot g_{k,t,\tau}(\widetilde{\boldsymbol{w}}_{k,t,\tau})$
    end for
    $\boldsymbol{w}_{k,t+1} = \widetilde{\boldsymbol{w}}_{k,t,N}$
    Call Algorithm 2 to get new mask $\boldsymbol{m}_{k,t+1}$
  end for
end for

图 4. DisPFL 算法过程 [3]

4 复现细节

4.1 与已有开源代码对比

DisPFL 开源代码： https://github.com/rong-dai/DisPFL
实验基于 DisPFL，针对原有模型个性化掩码更新与参数聚合两个阶段考虑改进方案。

- **个性化掩码更新（My-1）：** 原文方法主要基于 Top-K 保留策略与随机补充或依据本地数据获取梯度补充参数来进行调整。本实验则考虑将加权聚合阶段获得的全局模型权重用于指导掩码更新中 Re-grow 阶段的调整，以期进一步利用聚合节点提供的信息来指导模型得到更具有泛化性的结构（这一修改主要考虑稀疏度高的模型重叠结构少，不利于参数学习，也即参数聚合过程中从当前邻节点共享参数中获取信息少甚至没有）。

- **参数聚合策略（My-2）：** 同样考虑稀疏模型结构，在邻节点模型重叠参数少的情况下，通常的 Average 及 DisPFL 的掩码计数加权 Average（也可称 Element-wise Average）的方案并不能使本地模型有效地从该邻节点获取信息以提升本地模型的泛化性。本实验基于 My-1 进一步修改，将掩码更新调整到聚合阶段，在保留 Top-K 本地模型参数后再依据聚合模型参数进行 Re-grow 调整。

4.2 数据集及参数配置

- **模型**：ResNet-18
- **数据集**：CIFAR-10（Non-IID 处理：采用 Dirichlet 分布划分，$\alpha = 0.3$）
- **模型稀疏度**：0.5
- **对比算法**：FedAvg，DisPFL，修改的 My-1 和 My-2
- **模型性能指标**：基于本地测试数据集的准确率 Accuracy
- **通信网络结构**：time-varing 的随机网络（如图 2 (d)）
- 每两轮通信之间，本地训练轮次：5 epochs

5 实验结果分析

不同算法对比结果如图 5 所示。My-1 将聚合模型用于本地个性化掩码更新，在一定轮次后，对于聚合后模型的初始性能略优于原来的方案（My-1 vs DisPFL (after Aggregation)）不过对于再经过本地训练后的模型性能并没有明显提升（My-1 vs DisPFL (after Train)）。My-2 方案的效果相对于 DisPFL 则没有提升。从聚合后模型的初始性能上看，相对于 My-1 和 My-2 方案对模型掩码的定向调整，DisPFL 的随机调整策略在早期训练过程中表现较好，但在一定轮次后，由于余弦退火机制导致本地个性掩码的调整仍然较为剧烈，可能拖累模型的收敛速率。

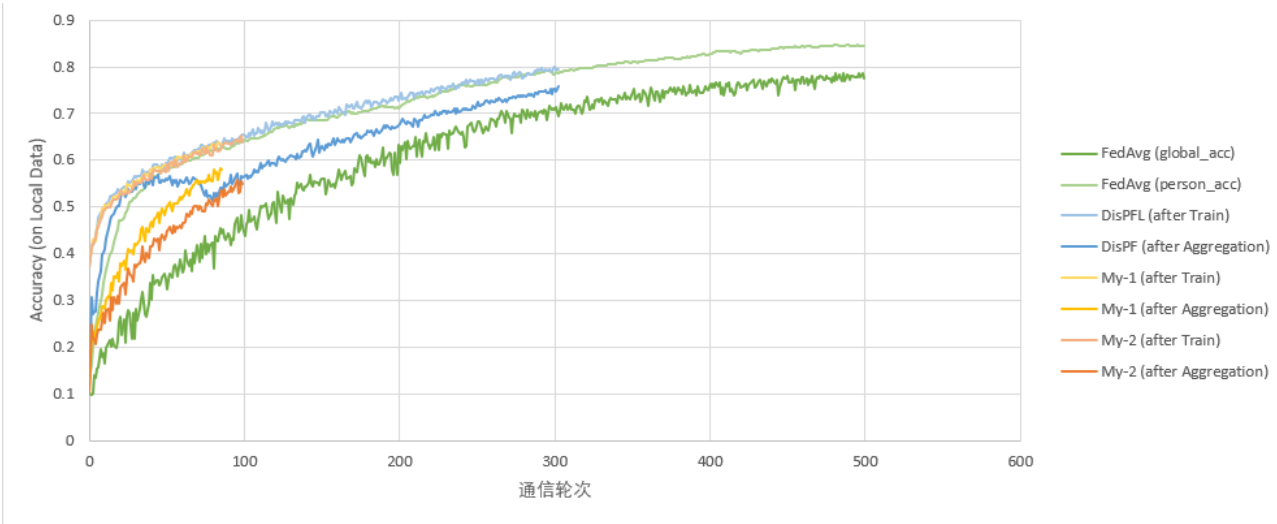

图 5. 不同算法在本地测试集上的准确度随通信轮次变化趋势

6 总结与展望

DisPFL 在个性化与去中心化的联邦学习上有不错表现，不过在模型聚合方案和个性化稀疏掩码方案上还存在一些问题。（1）全局信息聚合：由于稀疏模型间存在参数结构上的差异，只提取本地个性化稀疏掩码相关的信息（虽然个性化稀疏掩码在一定程度上也反映了本地数

据分布情况，如图 6 ，选择数据分布相一致的——即结构相一致的——邻节点作为主要学习对象也是合理的），无法充分利用到联邦学习的优势，在聚合选择的策略上可能有改进空间，尤其是当模型稀疏度高的情况，模型结构差异大，要找到重叠的结构较多的邻节点并不容易，这将导致模型能力从联邦学习中获得的提升变少，影响模型收敛速率和最终性能。在这一方面可以考虑借鉴联邦学习中客户端选择的相关研究，以及针对异构模型的聚合方案（常见的有知识蒸馏和迁移学习，但较为耗时），另外在 P2P 通信中也需要考虑异步联邦的问题。（2）此外，个性化稀疏掩码的调整较为粗糙，且基于余弦退火的调整过程需要较长时间才能稳定下来，也会影响模型收敛速率，优化这一剪枝和调整的过程也是可以考虑的方向。

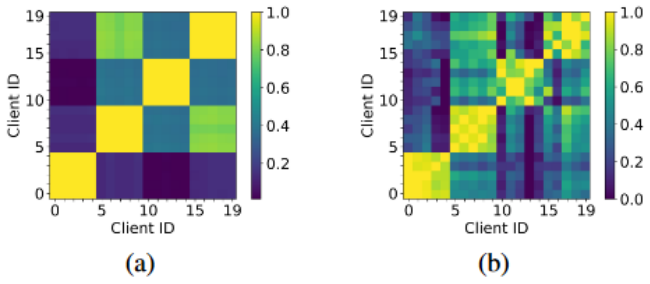

Figure 5. Explaining the learned masks, while (a) shows the cos-similarity of the training labels distributions between clients and (b) shows the aligned hamming distance between the learned masks.

图 6. 客户端间个性化稀疏掩码的相关性一定程度上也反映了数据集分布上的相关性 [3]

参考文献

[1] Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. *Advances in neural information processing systems*, 30, 2017.

[2] Debraj Basu, Deepesh Data, Can Karakus, and Suhas Diggavi. Qsparse-local-sgd: Distributed sgd with quantization, sparsification and local computations. *Advances in Neural Information Processing Systems*, 32, 2019.

[3] Rong Dai, Li Shen, Fengxiang He, Xinmei Tian, and Dacheng Tao. Dispfl: Towards communication-efficient personalized federated learning via decentralized sparse training. In *International Conference on Machine Learning*, pages 4587–4604. PMLR, 2022.

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

[5] Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the lottery: Making all tickets winners. In *International Conference on Machine Learning*, pages 2943–2952. PMLR, 2020.

[6] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020.

[7] Ang Li, Jingwei Sun, Xiao Zeng, Mi Zhang, Hai Li, and Yiran Chen. Fedmask: Joint computation and communication-efficient personalized federated learning via heterogeneous masking. In *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, pages 42–55, 2021.

[8] Daliang Li and Junpu Wang. Fedmd: Heterogenous federated learning via model distillation. *arXiv preprint arXiv:1910.03581*, 2019.

[9] 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.

[10] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

[11] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B Allen, Randy P Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 2020.

[12] Shiwei Liu, Tianlong Chen, Xiaohan Chen, Zahra Atashgahi, Lu Yin, Huanyu Kou, Li Shen, Mykola Pechenizkiy, Zhangyang Wang, and Decebal Constantin Mocanu. Sparse training via boosting pruning plasticity with neuroregeneration. *Advances in Neural Information Processing Systems*, 34:9908–9922, 2021.

[13] Hesham Mostafa and Xin Wang. Parameter efficient training of deep convolutional neural networks by dynamic sparse reparameterization. In *International Conference on Machine Learning*, pages 4646–4655. PMLR, 2019.

[14] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

[15] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[16] Tongtian Zhu, Fengxiang He, Lan Zhang, Zhengyang Niu, Mingli Song, and Dacheng Tao. Topology-aware generalization of decentralized sgd. In *International Conference on Machine Learning*, pages 27479–27503. PMLR, 2022.

[17] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pages 12878–12889. PMLR, 2021.