智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

基于动态阈值增强原型网络的联邦半监督学习模型

陈涛, 谢在鹏, 屈志昊

引用本文:
陈涛, 谢在鹏, 屈志昊. 基于动态阈值增强原型网络的联邦半监督学习模型[J]. 智能系统学报, 2024, 19(3): 534-545.
CHEN Tao, XIE Zaipeng, QU Zhihao. Federated semi-supervised learning model based on dynamic threshold enhanced prototype network[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(3): 534-545.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202311015

DOI: 10.11992/tis.202311015
网络出版地址: https://link.cnki.net/urlid/23.1538.TP.20240429.2051.002

基于动态阈值增强原型网络的联邦半监督学习模型

陈涛,谢在鹏,屈志昊
(河海大学 计算机与信息学院,江苏 南京 211100)

摘　要:目前,联邦半监督学习面临着有效利用训练过程中大量无标签数据的挑战。尽管通过轻量级的原型网络实现客户端之间的知识共享可以缓解伪标签质量问题,但仍然有瓶颈。本文提出一种新的动态阈值增强下的原型网络联邦半监督学习算法。通过引入课程伪标签技术,其核心是对不同类别样本的学习状态动态调整阈值,使模型能够学习高质量的样本,显著提高模型的预测性能。实验结果表明,本算法在多个数据集上均取得优异的测试性能。在 CIFAR-10 数据集上,本算法相对于同类算法至少提高 3% 的测试精度。此外在 SVHN 和 STL-10 数据集上也有 1%~7% 的领先优势。值得注意的是,本算法在处理异质性和同质性数据时表现出色,且对于不同比例的有标签和无标签数据都具有良好的适应性。本算法不仅提高测试精度,而且未带来额外的通信开销和计算成本。这些结果表明本算法在联邦半监督学习领域具有巨大潜力,并为实际应用提供了一个性能卓越且高效的解决方案。

关键词:联邦学习;半监督学习;知识共享;原型网络;伪标签;动态阈值;无标签数据;数据异质性
中图分类号:TP181　　**文献标志码**:A　　**文章编号**:1673−4785(2024)03−0534−12

中文引用格式:陈涛, 谢在鹏, 屈志昊. 基于动态阈值增强原型网络的联邦半监督学习模型 [J]. 智能系统学报, 2024, 19(3): 534–545.
英文引用格式: CHEN Tao, XIE Zaipeng, QU Zhihao. Federated semi-supervised learning model based on dynamic threshold enhanced prototype network[J]. CAAI transactions on intelligent systems, 2024, 19(3): 534–545.

Federated semi-supervised learning model based on dynamic threshold enhanced prototype network

CHEN Tao, XIE Zaipeng, QU Zhihao
(College of Computer and Information, Hohai University, Nanjing 211100, China)

Abstract: Currently, federated semi-supervised learning (FSSL) faces the challenge of making effective use of a large amount of unlabeled data during training. Although knowledge sharing between clients through a lightweight prototyping network can alleviate pseudo-label quality issues, there are still bottlenecks. In this paper, we propose a federated semi-supervised learning model based on dynamic threshold enhanced prototype network. By introducing Curriculum Pseudo labeling, the core is to dynamically adjust the threshold of the learning state of different classes of samples, so that the model can learn high-quality samples and significantly improve the prediction performance of the model. Experimental results show that our proposal has achieved excellent test performance on multiple datasets. On the CIFAR-10 dataset, our proposal improves the test accuracy by at least 3% compared with similar algorithms. In addition, there is a 1%~7% lead on SVHN and STL-10 datasets. It is worth noting that our proposal performs well in handling heterogeneous and homogeneous data, and has good adaptability to different proportions of labeled and unlabeled data. Our proposal can improve the test accuracy. What's more, it does not add additional communication overhead and computational cost. These results suggest that our proposal has great potential in the field of federated semi-supervised learning, and provides a high-performance and high-efficiency solution for practical applications.

Keywords: federated learning; semi-supervised learning; knowledge sharing; prototypical network; pseudo label; dynamic threshold; unlabeled data; heterogeneous data

近年来,联邦半监督学习成为一个新的研究课题,旨在利用无标签的数据来增强全局模型的开发。大多数现有的研究假设数据在客户端被完全标记。然而在实践中,有标签数据的数量通常

收稿日期:2023−11−13.　**网络出版日期**:2024−04−30.
基金项目:水灾害防御全国重点实验室"一带一路"水资源与可持续发展科技基金项目(2021490811);国家自然科学基金青年项目(62102131);江苏省自然科学基金青年项目(BK20210361).
通信作者:谢在鹏. E-mail: zaipengxie@hhu.edu.cn.

是有限的, 而且对数据标记非常昂贵。考虑到获取有标签数据成本极高, 而且客户端大部分数据是无标签的, 这种情况下如何利用少量有标签数据来提升模型性能成为了联邦半监督学习(federated semi-supervised learning, FSSL)的一个关键问题[1]。为了充分挖掘无标签数据的潜力, 许多研究者在 FSSL 中结合不同算法来提高模型性能。包括基于对比学习[2]、迁移学习[3-4]、集成学习[5]、知识蒸馏[6] 和原型学习[7] 等算法。本质都是通过知识共享来利用不同客户端的无标签数据。其中 ProtoFSSL[7] 利用原型网络作为知识共享的一个媒介, 基于多个原型网络对样本计算类概率值的平均聚合来生成伪标签。

然而客户端上原型网络训练受到客户端本地数据异质性的影响, 导致不同客户端上原型网络之间存在较大的梯度差异[8-9]。使得原型网络对无标签数据预测的类概率值不具有明显的区分度, 生成的伪标签具有较高的信息熵, 导致模型学习到错误的知识, 从而影响模型最终的预测性能。有研究提出直接锐化类概率值进行交叉熵计算[7], 然而没有经过阈值筛选的类概率值可能会产生不稳定的预测, 生成的伪标签可能会增加误标数据的风险, 导致模型在不同客户端上的泛化能力受限。

鉴于上述问题, 本文的动机是提出一种新的方法来解决 FSSL 下原型网络的模型漂移对伪标签质量产生的负面影响。为此引入课程伪标记技术, 核心是基于课程学习的动态阈值调整方法。与传统半监督学习中使用固定阈值筛选无标签数据的方法不同, 动态阈值增强下的原型网络联邦半监督学习算法(federated semi-supervised learning model based on dynamic threshold enhanced prototype network, FlexProtoFSSL)算法根据模型对每个类的学习状态动态调整阈值, 以选择高质量的样本进行学习。与现有方法相比, 本文具有以下贡献:

1) 动态阈值调整机制, 使得模型在不同时长根据学习状态选择合适的伪标签, 提高了模型学习的灵活性。

2) 课程伪标记技术的引入, 进一步提高伪标签的质量, 实现对未标记数据的有效利用。

3) 在不增加额外通信开销和计算负担的情况下, 模型在多个公认的基准数据集上取得的性能不亚于当下任何主流方法。

1　相关工作

1.1　联邦半监督学习

考虑到在本地客户端标记数据有限且花费昂贵, SSL 和 FL 场景很好地结合在一起, 产生了一个相对较新的问题, 称为联邦半监督学习[10]。FedMatch[11] 中介绍了 FSSL 的两种场景: 1) 标准场景。有标签数据和无标签数据均存放于本地, 服务器没有数据; 2) 不相交场景。有标签数据存放于服务器端, 无标签数据存放于客户端。

在 FSSL 领域中, FedMatch 方法[11] 首先提出一种分离学习方案, 将模型分为两组权值, 分别用于监督学习和无监督学习。对于无监督学习, 知识以模型权重的形式在客户端之间共享, 并使用客户端间一致性损失进行一致性正则化。但是, 单独训练两组参数并在客户端之间共享权重会增加客户端的计算量和通信开销。最近的一些工作集中在服务器有标签数据的场景。结合集成学习的算法中, FedFAME[4] 引入对比网络, 它由在线网和目标网组成, 在线网从训练数据中更新参数, 目标网络以动量机制缓慢更新并保留之前的训练信息, 可以解决数据异构问题。F2CMT[5] 结合集成学习思想, 提出局部模型的自我融合与不同客户端的跨模型融合来解决标签数据不足的问题, 可以提高模型的泛化能力且无需额外训练时间。但是计算复杂, 需要较大的计算开销。DS-FL[6] 融合了知识蒸馏和半监督学习思想, 将不同局部模型最后一层输出作为知识传输, 减少通信量, 同时通过减少熵值的聚合方法来提高模型对数据异质性的鲁棒性。ProtoFSSL[7] 对原型网络在 FSSL 中进行了扩展。客户端通过轻量级原型彼此共享知识而不使用模型权重来实现有效的客户间一致性正则化, 从而防止局部模型的发散。为了计算无标签数据的损失, 每个客户端基于共享原型创建伪标签, 与标记数据共同为局部模型训练提供指引。

相比之下, 本研究侧重于客户端有标签数据的标准场景, 深入研究了知识共享过程, 并提出一种新的基于原型网络提高伪标签质量的算法, 在数据异构下能提高模型的准确性和计算效率。

1.2　原型学习

原型学习[12] 就是训练一个原型网络模型, 该模型为每个类生成合理的原型(低维嵌入向量), 并利用其嵌入向量与每个原型之间的距离对新数据进行分类。原型概念已成功应用于元学习[13-14]、领域自适应[15-16] 等多个领域。最近尝试应用原型来解决客户端在 FSSL 设置中的数据异质性、通信开销、隐私保护等问题。Fedproto[17] 首先将原型引入联邦学习并取得了不错的效果。在本地构建标签类的原型网络, 送到全局进行聚合, 又重

新发送回本地进行训练。目的使局部数据的分类误差最小化，使得局部模型与全局模型足够接近。

尽管现有的工作在 FSSL 领域取得了显著成果，然而在面对复杂真实世界数据的情况下，仍然存在诸多挑战。本文通过提出在少量有标签数据和大量无标签数据的场景下，基于原型网络来实现客户端间知识共享，同时设置动态阈值保证原型网络对无标签数据预测的伪标签具有较高的准确度，最终提高模型的预测性能。

2　预备知识

2.1　原型网络

训练原型网络，为嵌入空间中的每个类提供良好的低维嵌入向量和原型。该网络将数据样本转换为嵌入向量，并使用与每个原型的距离对向量进行分类。定义 K 为类的集合，D_k 为类 k $(k \in K)$ 的训练数据集，训练集 D_k 分为两部分，支持集 S_k（D_k 的随机子集）以及剩余的查询集 $Q_k(D_k \backslash S_k)$。然后，根据支持集计算类 k 的原型 $\boldsymbol{c}_k$ 为

$$\boldsymbol{c}_k = \frac{1}{|S_k|}\sum_{x \in S_k} \boldsymbol{f}_{\boldsymbol{\theta}}(x^u) \tag{1}$$

式中：x^u 表示无标签数据，$\boldsymbol{f}_{\boldsymbol{\theta}}$ 表示由权重 $\boldsymbol{\theta}$ 参数化计算获取低维嵌入向量。然后，用损失函数对模型进行训练：

$$L = -\sum_{k \in K}\sum_{x \in Q_k} \log \frac{\exp(-d(\boldsymbol{f}_{\boldsymbol{\theta}}(x^u), \boldsymbol{c}_k))}{\sum\limits_{k' \in K} \exp(-d(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x^u}), \boldsymbol{c}_{k'}))} \tag{2}$$

其中 d 是欧氏距离函数。对模型进行训练使同一类的嵌入向量位置靠得近，不同类的嵌入向量位置靠得远。

2.2　无标签数据的伪标记

伪标签是一种半监督学习技术，它通常用于在训练数据有限的情况下改进模型性能。模型首先使用已有的有标签数据进行训练，然后使用该模型对未标记的数据进行预测，并为这些未标记的数据分配一个伪标签。这些伪标签通常是模型预测的最高概率类别。基于一致性正则化的伪标签对于有效的半监督学习是非常重要的。例如在 MixMatch[18] 中，一些用于一个无标签数据 x^u 的增强数据 $x_a^u (a = 1,2,\cdots,A)$ 用来制作一个伪标签。为此计算模型在数据 x^u 的 A 个增强数据上预测的平均值 $p(x^u)$。MixMatch[18] 进一步锐化 $p(x^u)$ 来降低标签预测的信息熵：

$$\bar{p}_k(x^u) = \frac{p_k(x^u)^{\frac{1}{T}}}{\sum\limits_{k' \in K} p_{k'}(x^u)^{\frac{1}{T}}} \tag{3}$$

式中：$p_k(x^u)$ 是 x^u 关于类 k 的概率，T 是一个称为温度的超参数。锐化概率分布 $\bar{p}_k(x^u)$ 是数据 x^u 软化的伪标签。

3　基于原型网络的动态阈值联邦半监督学习

3.1　问题定义

考虑 FSSL 的标准场景下，服务器端是没有数据的，每个参与的客户端 $i \in M_r$ 都有一个 Non-IID 的私有数据集 $D_i = D_i^l \cup D_i^u$，即有标签数据和无标签数据。$D_i^l = \{(x_1^l, y_1^l),(x_2^l,y_2^l),\cdots,(x_n^l,y_n^l)\}$ 为 n 个有标签数据集合。$D_i^u = \{(x_1^u,y_1^u),(x_2^u,y_2^u),\cdots,(x_m^u,y_m^u)\}$ 为 m 个无标签数据集合。尤其在 $m \gg n$ 时这类问题很值得研究。在这项工作中，主要关注无标签数据，需要计算无标签数据 x_i^u $(i \in m)$ 的伪标签 $\hat{y}_i$，计算方法为

$$\hat{y}_i = \underset{k \in K}{\operatorname{argmin}} \sum_{i=1}^{m} d(\boldsymbol{f}_{\boldsymbol{\theta}}(x_i^u), \boldsymbol{c}_k) \tag{4}$$

目标是利用 $\{(D_1^l \cup D_1^u),(D_2^l \cup D_2^l),\cdots,(D_{|M_r|}^l \cup D_{|M_r|}^l)\}$ 来生成一组原型网络模型参数 $\{\boldsymbol{\theta}_1, \boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_k\}$，可以最小化在带有伪标签的无标签数据 $\{x_1^u,x_2^u,\cdots,x_m^u\}$ 上的经验损失，模型参数 $\boldsymbol{\theta}^*$ 计算方法为

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,\cdots,\boldsymbol{\theta}_k\}}{\operatorname{argmin}} \sum_{i=1}^{m} \operatorname{Loss}(\hat{y}_i, \bar{p}(y_i;\boldsymbol{\theta})) \tag{5}$$

式中：Loss 函数可以是任何形式的监督学习损失函数，伪标签 $\hat{y}_i$ 即是模型预测数据所属最确定类别的标签，$\bar{p}(y_i;\boldsymbol{\theta})$ 是在模型 $\boldsymbol{\theta}$ 上对数据 y_i 预测的类概率值。通过最小化无标签数据的伪标签和预测类概率值的交叉熵损失获得最优化模型参数 $\boldsymbol{\theta}^*$。

本工作需要在每个客户端为每个类训练一个原型网络。在每个全局通信回合中，对于客户端的局部迭代，每个参与的客户端使用其本身的有标签和无标签数据集更新本地模型参数，并更新客户端上每个类的原型网络[19]。在每个局部迭代中，客户端从标签数据中为每个类随机采样一个有标签支持集，并从有标签数据中除有标签支持集以外的数据中为每个类中随机采样一个有标签查询集，同时从无标签数据中随机采样一个无标签查询集，需要同时利用这 3 类数据集进行一致性正则化。在伪标签的生成过程中，可以根据模型的输出概率分布来计算每个样本的信息熵，并将信息熵用作衡量预测类概率区分度的指标。由于数据异质性导致客户端模型漂移，全局聚合的类概率值往往具有较高的信息熵，导致伪标签准确度不高。可以在训练时候设置一个阈值，筛选出高于阈值的类预测值来制作伪标签，这样可以

确保伪标签准确度较高, 从而提高模型的性能。

3.2　基于动态阈值的课程伪标签

由于数据异质性导致每个客户端上每个类的学习状态或学习效果不尽相同, 模型在训练过程中使用固定阈值筛选伪标签样本不能够很好地适应数据和模型变化, 容易导致训练过拟合。因此通过动态阈值筛选合适的伪标签样本进行学习[20]很有必要。根据模型中每个类的学习状态来动态调整阈值。这种策略确保伪标签质量的同时逐步提高模型性能。

在训练初始阶段使用较低的阈值帮助模型先从数据中捕捉到最简单和直观的特征, 然而模型在初始阶段的参数初始化存在确认偏差, 导致对无标签数据产生错误的伪标签, 从而学习到错误的知识, 导致模型的学习状态不稳定且不可靠。为了解决这一问题, 引入阈值预热的方法, 可以帮助模型在训练早期学习简单样本, 更好地利用无标签数据, 稳定地提高模型的学习状态[21-23]。在阈值预热过程中, 通过逐渐提高所有类别的阈值来解决训练早期阶段出现大量错误伪标签的问题, 这样可以在训练的早期阶段创造一个学习热潮[24], 使大部分无标签数据得到利用。

随着模型不断学习迭代, 模型需要更加确定才能将样本归类, 因此逐渐提高阈值, 通过这种适应性调整, 模型具备对复杂化问题的适应和学习能力。阈值很高时, 一个类的学习状态或学习效果可以根据预测的类概率最大值 (即模型对其最确定的类别的概率) 达到阈值的样本数量来衡量。当某一类别样本的最高类概率预测值很高, 比如 0.96, 大于动态阈值, 意味着模型对该类样本的分类非常自信, 表示该类别样本的学习难度较小, 学习状态较好; 相反, 当某一类别样本的最高类概率预测值较低, 比如 0.50, 小于动态阈值, 意味着模型对该类样本的分类不那么确定, 表示该类别样本的学习难度较大, 学习状态较差。当某一类别样本的学习难度大时适当降低阈值, 保证模型对困难样本的学习, 从而提高性能并保持良好的泛化能力。这种方法可以帮助模型动态调整阈值, 高阈值过滤掉有噪声的伪标签, 只留下高质量的伪标签[25], 可以大大减少确认偏差, 从而更好地学习无标签数据。设计的动态阈值缩放函数为[24]

$$\sigma_i^t(k)=\sum_{j=1}^{|D_i^u|}1\left(\max_k\left(p_{i,j}(x^u)\right)>\tau\right)\cdot 1\left(\operatorname*{argmin}_k\left(p_{i,j}(x^u)\right)=k\right) \tag{6}$$

式中: $p_{i,j,k}(x^u)$表示客户端 i 上第 j 个无标签数据在辅助客户端的原型网络上计算得到的类概率值; max 函数表示获取类概率值中最大的概率值; argmax 函数表示寻找使得类概率取得最大值所属的类; 1 表示指示函数[24], 对于满足条件的函数值为 1, 否则为 0。当阈值τ确定时, 一个类的学习状态可以通过预测落在该类中并且类概率最大值高于阈值的样本数量来反应。其中 $\sigma_i^t(k)$反映了第 i 个客户端上第 k 类数据在 t 时刻的学习状态。

$$\beta_i^t(k)=\frac{\sigma_i^t(k)}{\max_k\sigma_i^t(k)} \tag{7}$$

$$\mathcal{T}_i^t(k)=\beta_i^t(k)\cdot\tau \tag{8}$$

$\beta_i^t(k)$通过对$\sigma_i^t(k)$应用归一化, 使其范围在 0~1, 然后通过它缩放固定阈值τ得到动态阈值$\mathcal{T}_i^t(k)$。请注意在这个过程中, 如果无标签数据被归为不正确的类, 阈值最后也会相应降低。

在阈值预热阶段, 阈值计算过程为

$$\beta_i^t(k)=\frac{\sigma_i^t(k)}{\max\left\{\max_k\sigma_i^t(k),m-\sum_k\sigma_i^t(k)\right\}} \tag{9}$$

其中$m-\sum_k\sigma_i^t(k)$表示无标签数据的个数, 这确保了在训练初期阶段所有估计的学习状态从 0 逐渐上升, 直到未使用的无标签数据集的数量不占主导地位。这个周期的持续时间取决于数据集的无标签数据集的数量 (m) 和学习难度 ($\sigma_i^t(k)$的增长速度)。

以上设计的动态阈值容易计算, 不需要引入额外的推理过程, 也不需要额外的验证集。由于模型初始阶段学习状态不理想, 动态阈值往往不高, 从而鼓励该类的更多简单样本参加训练, 随着模型不断迭代训练, 模型对类样本逐渐适应, 类概率最大值达到阈值的样本数量递增, 同时类概率值可能具有更高的区分度, 伪标签的信息熵也更低, 动态阈值过滤掉有噪声的伪标签, 只留下高质量的伪标签, 从而提高模型预测精度。这些通过筛选的样本及其标签类别都会被标记, 并在下一个时间步长被重新计算生成新的动态阈值, 当然如果标签的预测准确率下降, 表明类的学习状态越不令人满意, 就会导致动态阈值降低, 随着模型学习状态的变化逐步引入学习样本的学习策略就是课程伪标签技术[26-27]。

3.3　动态阈值增强的原型网络联邦半监督学习

在每次局部训练中, 辅助客户端 (H_r) 会使用有标签支持集来训练本地原型网络。活跃客户端 (M_r)使用辅助客户端中的外部原型对无标签的查询数据x^u上计算类概率分布, 每个查询数据在每个辅助客户端都会得到类概率值。在每个客户

端的局部训练轮次中，客户端首先使用局部权重为无标签的数据x^u计算嵌入向量$\boldsymbol{f}_{\boldsymbol{\theta}}(x^u)$。然后计算类 k 和辅助客户 j 上嵌入向量$\boldsymbol{f}_{\boldsymbol{\theta}}(x^u)$与原型网络$\boldsymbol{c}_{j,k}$之间的欧氏距离。客户端 i 上无标签数据x^u经辅助客户端 j 的原型网络计算属于类 k 的概率值 $p_{i,j,k}(x^u)$，计算公式为

$$p_{i,j,k}(x^u)=\frac{\exp(-\mathrm{d}(f_{\boldsymbol{\theta}}(x^u),\boldsymbol{c}_{j,k}))}{\sum_{k'\in K}\exp(-\mathrm{d}(f_{\boldsymbol{\theta}}(x^u),\boldsymbol{c}_{j,k'}))} \tag{10}$$

根据动态阈值筛选类概率值并计算均值 $p_{i,k}(x^u)$，最后锐化以降低伪标签的信息熵[28]，计算最终类概率值$\bar{p}_i(x^u)$和伪标签$\hat{y}_i$。

在对无标签查询集数据进行伪标记后，客户端使用有标签查询数据和无标签查询数据计算总的损失。最后使用CroE函数计算每个局部训练轮次的交叉熵损失。本文将 FlexProtoFSSL 算法中的损失表示为有监督和无监督损失的加权组合 $L_{\text{total}}=L_{\text{s}}+\lambda\cdot L_{\text{u}}$。其中$L_{\text{s}}$为客户端 i 上有标签数据的监督损失：

$$L_{\text{s}}=\frac{1}{\sum_{k\in K}|Q_{i,k}^{L}|}\sum_{k\in K}\sum_{x^l\in Q_{i,k}^{L}}\mathrm{CroE}\left(p_{i,k}\left(x^l\right),y_i\right) \tag{11}$$

L_{u}为客户端 i 上无标签数据的无监督损失：

$$L_{\text{u}}=\frac{1}{|Q_i^U|}\sum_{x^u\in Q_i^U}\mathrm{CroE}(\bar{p}_i(x^u),\hat{y}_i) \tag{12}$$

3.4　算法细节

在本节中，描述了 FlexProtoFSSL 算法细节。它旨在通过使用轻量级原型作为客户端间知识，基于课程学习制作动态阈值，从原型预测的结果中筛选出达标的制作伪标签，有效解决了 FSSL 中标签数据不足的问题。图 1 给出了算法的具体流程。

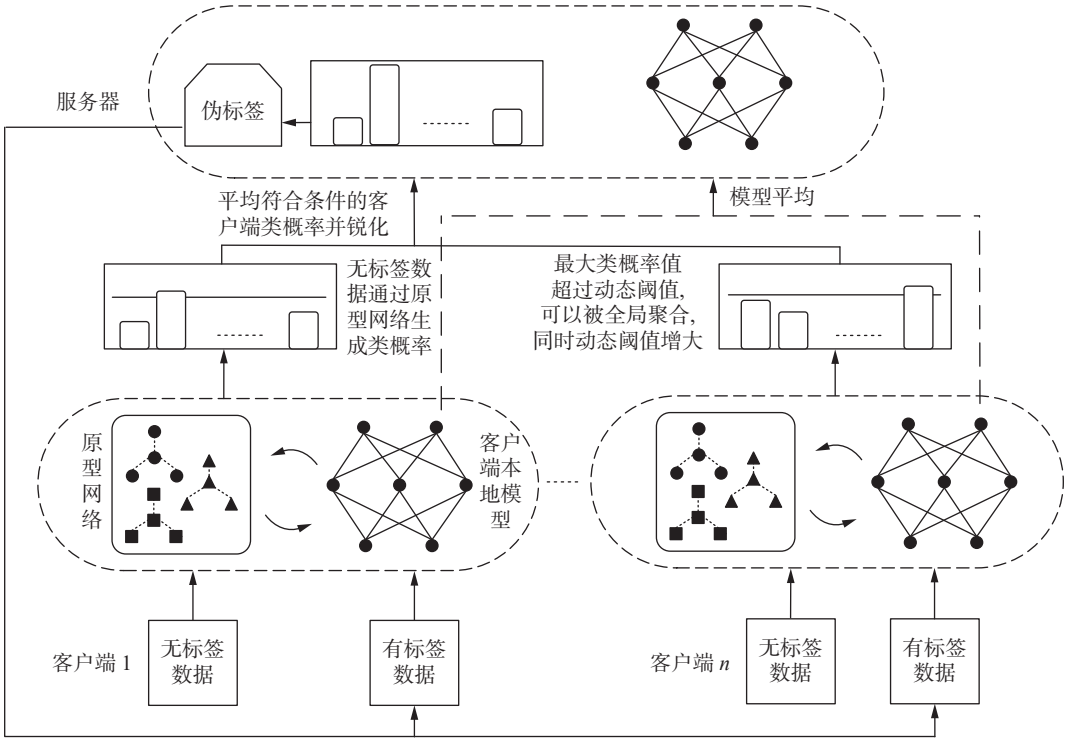


图 1　FlexProtoFSSL 算法流程

Fig. 1　FlexProtoFSSL algorithm flow

本工作的目标是以联邦方式训练一个由θ参数化的原型网络$\boldsymbol{f}_{\boldsymbol{\theta}}$。在每个通信回合 r 中，对于局部训练轮次 e，参与的客户端$i\in M_r$使用其有标签和无标签的数据集局部更新全局参数$\boldsymbol{\theta}_i$，并为每个类更新其局部原型$\boldsymbol{c}_{i,k}$。在每个局部训练轮次 e 中，客户端随机从有标签数据集中抽取一个有标签支持集来训练本地模型以及原型网络，从有标签数据集中除有标签支持集以外的数据中抽取一个有标签查询集，并从无标签数据集中随机抽取一个无标签查询集进行一致性正则化。现在客户端对本地模型和原型网络训练一定轮次，达到一个初始化预热过程。这确保了无标签数据的数量不占主导地位，然后计算动态阈值，无标签查询集在不同辅助客户端的原型网络上计算类概率值，并筛选出合格的类概率分布，然后对该数据的不同类概率分布取平均，最后锐化类概率值生成伪标签。

基于原型网络的动态阈值联邦半监督学习算法步骤如下：

全局训练阶段：

1) 参数初始化, 其中包括有标签样本集D^L和无标签样本集D^U、客户端模型初始化、原型网络初始化、初始阈值、辅助客户端数量等参数[29]。

2) 客户端从有标签样本集中随机选取一部分样本作为支持集, 训练本地模型和原型网络, 服务器根据客户端模型聚合生成全局模型。

客户端本地训练阶段:

3) 从无标签样本集中随机选取一部分样本作为无标签查询集, 在本地模型对其计算预测值, 从不同辅助客户端中筛选出预测值高于阈值的样本, 全局聚合处理后生成伪标签。

4) 根据样本落入该类的数量计算每个类的估计学习状态, 归一化得到最终阈值。无标签样本中预测类概率的最大值若高于阈值则从无标签样本集中剔除, 加入有标签样本集。

5) 客户端从有标签数据集和无标签数据集中各选取一部分作为查询集, 在本地模型上对有标签查询集预测与真实标签计算有标签的交叉熵损失, 无标签查询集的预测值与伪标签计算无标签的交叉熵损失[30]。

6) 客户端根据有标签数据集训练局部原型网络, 并将局部模型发送至服务器。

具体的算法流程见算法 1 FlexProtoFSSL。

算法 1 FlexProtoFSSL

客户端执行:

输入　全局模型参数$\boldsymbol{\theta}^{r-1}$, 辅助客户端上的类原型网络$\{\boldsymbol{c}_{j,k}\}_{j\in H_r,1\leqslant k\leqslant K}$;

输出　客户端本地模型参数$\boldsymbol{\theta}_i^r$, 客户端上的本地类原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

1) 初始化客户端本地模型$\boldsymbol{\theta}_{i,0}^r \leftarrow \boldsymbol{\theta}^{r-1}$;

2) 选择有标签支持集$S_{i,k}^L$、有标签查询集$Q_{i,k}^L$、无标签查询集Q_i^U和原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

3) 进行阈值预热;

4) FOR 本地训练轮次 $e = 1, 2, \cdots, E$;

5) FOR 类 $k = 1, 2, \cdots, K$;

6) 计算$\sigma_t(k)$、$\beta_t(k)$、$\mathcal{T}_t(k)$;

7) 利用本地模型$\boldsymbol{\theta}_{i,e-1}^r$和有标签支持集$S_{i,k}^L$训练每个类的原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

8) 重复 6) ~ 7);

9) 利用辅助客户端类原型网络计算无标签查询集的类概率值$p_{i,k}(x^u)$;

10) 判断 $\max(\{p_{i,k}(x^u)\}_{k\in K}) > \mathcal{T}_t(k)$, 条件满足进行伪标记, 否则返回无标签数据集;

11) 更新本地模型: $\boldsymbol{\theta}_{i,e}^r \leftarrow \boldsymbol{\theta}_{i,e-1}^r - \eta\nabla_{\boldsymbol{\theta}}\text{Loss}$;

12) 重复 5) ~ 11) E 次;

13) 利用本地模型$\boldsymbol{\theta}_{i,E}^r$和有标签支持集$S_{i,k}^L$再次训练每个类的原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

14) 将更新后的本地模型$\boldsymbol{\theta}_i^r$和原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$发送给服务器;

15) END

服务器端执行:

输入　客户端本地模型参数$\boldsymbol{\theta}_i^r$, 客户端上的类原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

输出　全局模型参数$\boldsymbol{\theta}^r$, 客户端上的类原型网络$\{\boldsymbol{c}_{i,k}\}_{1\leqslant k\leqslant K}$;

1) 初始化全局模型$\boldsymbol{\theta}^0$;

2) FOR 全局训练轮次 $r = 1, 2, \cdots, R$;

3) 从M_r中随机选择 m 个客户端;

4) 从上一轮活跃客户端M_{r-1}中选择辅助客户端集合H_r;

5) FOR 客户端 $i \in M_r$并行执行;

6) 获得客户端新更新的本地模型$\boldsymbol{\theta}^r$;

7) 客户端执行原型网络$\{\boldsymbol{c}_{j,k}\}_{1\leqslant k\leqslant K, j\in H_r}$更新;

8) 重复 6) ~ 7), 利用客户端模型$\{\boldsymbol{\theta}_i^r\}_{i\in M_r}$更新全局模型$\boldsymbol{\theta}^r$;

9) 存储客户端原型网络$\{\boldsymbol{c}_{i,k}\}_{i\in M_r,1\leqslant k\leqslant K}$;

10) 服务器发送更新后的全局模型$\boldsymbol{\theta}^r$和原型网络$\{\boldsymbol{c}_{i,k}\}_{i\in M_r,1\leqslant k\leqslant K}$给客户端;

11) 重复 3) ~ 9) R 次;

12) END

4　实验结果及分析

主要对算法 FedAvg[31]、FedProx[32]、FedMatch[11]、FixMatch[1]、RSCFed[33]、FedRGD[34] 和 ProtoFSSL[7] 在常见的 3 类数据集 CIFAR-10、SVHN 和 STL-10[35] 上比较测试性能。FedAvg[31] 和 FedProx[32] 应用于受限的 SL 场景, 每个客户端只有小部分的有标签数据且没有任何无标签数据。

为了公平比较, 遵循 FSSL 其他研究中的相同设置。共 100 个客户端, 每轮随机抽取 5 个活跃客户 ($|H_r|$=5) 进行训练。使用 ResNet9 模型。其中 CIFAR-10 数据集中有 54 000 个训练集, 3 000 个验证集和 3 000 个测试集。SVHN 数据集中有 54 000 个训练集, 2 000 个验证集和 2 000 个测试集。训练数据均匀分布到 100 个客户端(即每个客户端 540 个数据)。有标签数据和无标签数据的比例是固定的, 其中每个类包含 5 个有标签数据样本, 其余 490 个是无标签数据样本。对于客户端的无标签数据设置分为两种情况: 一种是独立同分布的, 即每个客户端上每个类具有相同数量的数据; 还有一种就是非独立同分布的, 即每

个客户端具有不平衡的类分布。对于 STL-10, 每个客户端有 1 080 个数据样本, 其中 100 个数据有标签的, 剩余 980 个是无标签的数据。

4.1　实验结果

表 1 对 FlexProtoFSSL 算法进行了细致的实验研究, 以考察其在 IID 和 Non-IID 场景下的性能与优势。实验共划分为 3 个大组, 分别在 CIFAR-10、SVHN 及 STL-10 数据集上执行。在每个大组实验中, 分别在对应数据集的 IID 版本和 Non-IID 版本上训练, Non-IID 的程度被设定为$\alpha = 0.01$。这样共计形成了 6 个小组实验。不仅针对所提出的 FlexProtoFSSL 算法进行了详尽的评估, 而且每一大组实验中还纳入对照的 FedMatch[11]、FixMatch[1]、RSCFed[33]、FedRGD[34] 和 ProtoFSSL[7] 算法。这样的对比旨在全面验证 FlexProtoFSSL 算法在提升全局模型的预测准确性能方面的有效性和优越性, 并展现其对不同数据集在不同数据分布情况的动态自适应能力。

表 1　不同数据集的测试精度与其他方法的比较

Table 1　Comparison of test accuracy to other methods from different datasets　%

方法	CIFAR-10		SVHN		STL-10	
	IID	Non-IID	IID	Non-IID	IID	Non-IID
FedAvg	62.2	—	83.5	—	72.3	—
FedProx	62.7	—	83.9	—	72.0	—
FedMatch	62.5	61.6	85.5	86.0	73.7	72.7
FixMatch	63.4	61.8	86.6	86.5	73.6	71.6
RSCFed	62.4	62.1	79.9	79.1	75.8	75.1
ProtoFSSL-FedAvg	66.3	65.5	87.7	87.1	76.6	75.3
ProtoFSSL-FedProx	66.7	66.3	87.9	87.8	76.9	75.8
FedRGD	66.7	64.3	—	—	—	—
FlexProtoFSSL-FedAvg	69.6	68.1	88.3	87.7	78.4	78.0
FlexProtoFSSL-FedProx	70.5	69.3	89.2	88.2	79.2	78.2
FlexProtoFSSL(with BN)-FedAvg	71.7	71.9	89.8	89.4	80.7	80.4
FlexProtoFSSL(with BN)-FedProx	72.2	71.5	89.7	89.1	80.9	80.5

在 CIFAR-10 数据集上, 当数据集是 Non-IID 时, FlexProtoFSSL 与对照算法均产生了准确度下降的情况。与 IID 时相比, FedMatch 准确度下降 0.9 个百分点; FixMatch 准确度下降 1.6 个百分点; RSCFed 准确度下降 0.3 个百分点, ; ProtoFSSL-FedAvg 准确度下降 0.8 个百分点; ProtoFSSL-FedProx 准确度下降 0.4 个百分点; FedRGD 准确度下降 2.4 个百分点; FlexProtoFSSL-FedAvg 准确度下降 1.5 个百分点; FlexProtoFSSL-FedProx 准确度下降 1.2 个百分点。

在 SVHN 数据集上, 当数据集是 Non-IID 时, FlexProtoFSSL 与对照算法大多产生了准确度下降的情况。与 IID 时相比, FixMatch 准确度下降 0.1 个百分点; RSCFed 准确度下降 0.8 个百分点, ProtoFSSL-FedAvg 准确度下降 0.6 个百分点; ProtoFSSL-FedProx 准确度下降 0.1 个百分点; FlexProtoFSSL-FedAvg 准确度下降 0.6 个百分点; FlexProtoFSSL-FedProx 准确度下降 1.0 个百分点; 只有 FedMatch 准确度上升 0.5 个百分点。

在 STL-10 数据集上, 当数据集是 Non-IID 时, FlexProtoFSSL 与对照算法均产生了准确度下降的情况。与 IID 时相比, FedMatch 准确度下降 1.0 个百分点; FixMatch 准确度下降 2.0 个百分点; RSCFed 准确度下降 0.7 个百分点; ProtoFSSL-FedAvg 和 ProtoFSSL-FedProx 准确度分别下降 1.3 个百分点和 1.1 个百分点; FlexProtoFSSL-FedAvg 和 FlexProtoFSSL-FedProx 准确度分别下降 0.3 个百分点和 0.4 个百分点。

在 3 个数据集上 FlexProtoFSSL 算法对全局模型准确度的优化效果更为突出。虽然这个算法在 Non-IID 场景下准确度会下降, 但依然维持着良好的泛化性能。同时在 IID 和 Non-IID 场景下

都获得最高的准确度。实验表明 FlexProtoFSSL 算法在全局模型泛化性能优化方面的有效性与优越性, 也表明对于不同数据分布具有鲁棒性。

4.1.1　准确性

3 种数据集在不同数据分布下的准确度如图 2 所示, 本文算法在 3 个数据集的 IID 场景下准确度均高于 FedAvg[31] 算法和 FedProx[32] 算法, 表明本文算法可通过动态阈值筛选的方法从客户端间原型网络上提取有用信息, 制作高置信度的伪标签, 从而帮助模型高质量训练。本文算法性能显著超过 FedMatch[11] 算法, 在 CIFAR-10 和 STL-10 数据集上最为明显, 准确度高 6 个百分点左右。FedMatch 引入基于伪标签和预测类概率值计算交叉熵损失的一致性正则化技术, 表明动态阈值增强的原型网络进行知识共享的优越性, 可更好地获得客户端间的知识, 提高模型的泛化性能。

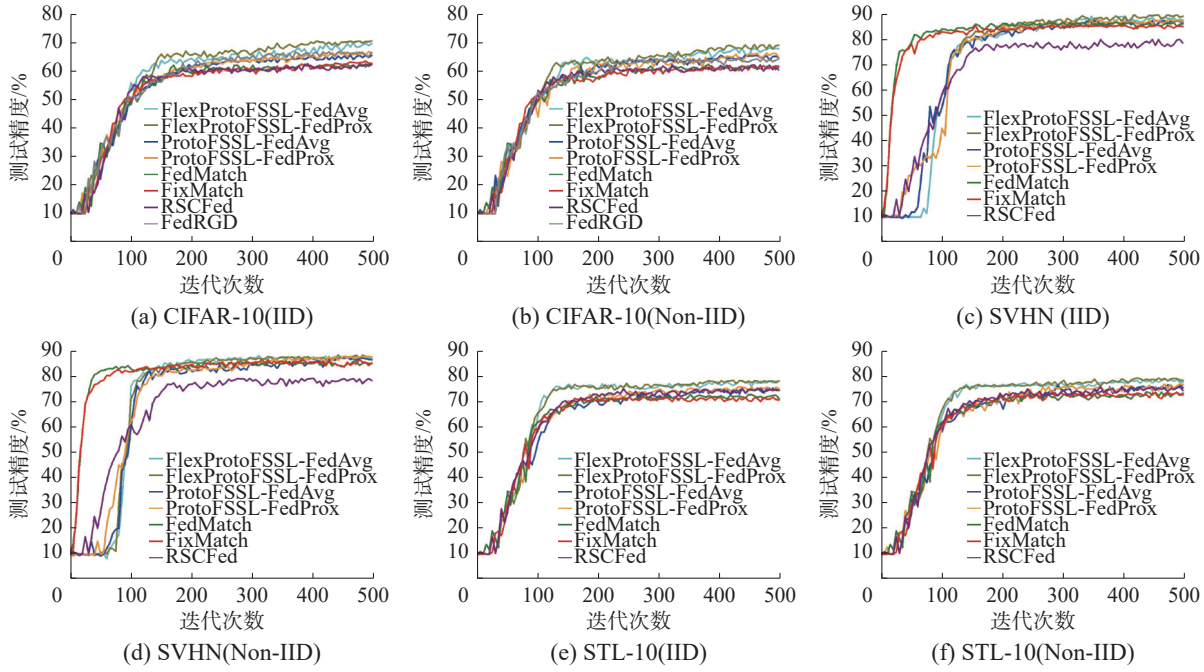


图 2　3 种数据集在不同数据分布下的准确度

Fig. 2　Accuracy of the three data sets under different data distributions

FixMatch[1] 算法是对图像数据的弱增强版本和强增强版本进行一致性正则化, 强弱图像的预测类概率分布没有一致, 而是根据弱增强数据生成的高置信度的伪标签, 结合强增强图像的输出类概率计算交叉熵损失, 进行一致性正则化。FlexProtoFSSL 算法在 3 个数据集上的性能全面超越 FedMatch[11] 算法, 即使在 SVHN 数据集上准确度也高 2 个百分点左右, 这表明动态阈值增强后可以获取高置信度的伪标签, 帮助模型学习到有用的知识, 从而提高模型准确度。

FlexProtoFSSL 算法在 3 个数据集上的准确度全面超越利用随机抽样子模型获取共识的 RSCFed[33] 算法, 表明动态阈值增强的原型网络联邦半监督学习模型性能优于基于模型共享的随机抽样获得的共识模型。不共享客户端模型的知识共享也可以获得很好的泛化性能。

FedRGD[34] 算法提出采用一致性正则化、批量归一化以及基于分组的模型平均技术减少梯度多样性, 提高了模型的预测准确度, 尤其在 IID 下取得最好的性能, 但仍逊色于 FlexProtoFSSL 算法 3 个百分点以上的精度表现。

在 3 个数据集上 FlexProtoFSSL 算法均高于 ProtoFSSL[7] 算法准确度 1~3 个百分点, 表明基于原型网络的联邦半监督学习中引入动态阈值确实能够提高伪标签的置信度, 更有效利用无标签数据来提高模型的泛化性能。

算法 FlexProtoFSSL 在实验中, 当数据集是 Non-IID 时, 模型性能略逊于在 IID 下数据集 1~2 个百分点, 表明动态阈值增强的原型网络联邦半监督学习模型可以有效地防止了客户端间模型偏移导致的泛化性能影响。在 SVHN 中, FlexProtoFSSL 甚至可以与完全监督学习相媲美。如上图所示, 在模型训练轮次达到 500 轮时, FlexProtoFSSL 测试结果趋于平稳达到收敛条件且准确度都高于其他算法。

将批量归一化 (BN) 方法[36] 应用于算法 FlexProtoFSSL, 观察到 FlexProtoFSSL 在数据集上测试性能有所提升。这表明使用动态阈值增强技术

与使用 BN 来解决局部模型梯度多样性技术是互补的，都可以提高模型泛化性能。

最后进行有标签与无标签数据比例对测试精度的影响的实验。如表 2 通过调整 CIFAR-10 数据集中有标签数据与无标签数据的比例测试不同算法的预测性能。结果显示 RSCFed[33] 算法具有最低测试精度，比不使用无标签数据的 FedAvg[31] 性能更差，表明利用无标签数据的方式可能会降低有标签数据的训练。

此外，FixMatch-FedAvg[1] 在少数情况下表现不如 FedAvg[31]。这意味着 FL 和 SSL 方法的简单组合并不一定会通过使用额外的无标签数据来提高性能。另一方面，FlexProtoFSSL 几乎在所有情况下的准确率都高于 ProtoFSSL[7]，显示了动态阈值增强下利用原型网络进行知识共享的有效性。

表 2　不同算法的不同比例的有标签和无标签数据准确性

Table 2　Test accuracy of various methods with different proportions of labeled and unlabeled data　%

算法	数据集比例			
	100∶440		400∶140	
	IID	Non-IID	IID	Non-IID
FedAvg	67.9	67.5	77.2	77.0
FixMatch-FedAvg	68.4	68.6	77.6	76.7
FedMatch	65.5	64.9	75.5	75.3
RSCFed	69.3	68.7	77.4	77.1
ProtoFSSL-FedAvg	72.6	72.5	79.3	79.0
FlexProtoFSSL-FedAvg	72.5	73.1	80.5	79.9

4.1.2　计算和通信成本

由于边缘计算场景中客户端资源受限，减少算法的计算和通信开销是很有必要的。假设 Θ 表示模型的大小，F 是一个数据样本运行在模型上的计算成本，C 表示原型大小。为了公平比较，假设所有算法都是在一个局部训练轮次中消耗整个有标签和无标签数据集。表 3 给出了使用模型 ResNet9 在数据集 CIFAR-10 上比较该算法与 MixMatch[18] 和 FxiMatch[1] 在计算和通信方面开销。

表 3　每轮中的每个客户端计算和通信成本

Table 3　Per-client computation and communication costs in each round

方法	计算成本	通信成本
MixMatch-FedAvg	$F(\lvert D_i^l\rvert+A\lvert D_i^u\rvert)E$	2Θ
FixMatch-FedAvg	$F(\lvert D_i^l\rvert+2\lvert D_i^u\rvert)E$	2Θ
FedRGD	$F(\lvert D_i^l\rvert+2\lvert D_i^u\rvert)E$	2Θ
FlexProtoFSSL	$C\lvert H_r\rvert\lvert K\rvert\lvert D_i^u\rvert E+F\lvert D_i^l\rvert$	$C(\lvert H_r\rvert+1)\lvert K\rvert$

在计算成本分析中，MixMatch[18] 的成本随着数据增强的数量和辅助客户端数量的增加而增加，这使得该算法难以扩展，FixMatch[1] 也是一样的计算开销。FlexProtoFSSL 提供了最低的计算成本，因为它不考虑数据增强且只做原型距离计算而不是全量模型的计算。在通信成本方面，所有算法至少需要 2Θ 才能在服务器和客户端之间交换全局和本地模型，只有 FlexProtoFSSL 是通信成本最小的，取决于原型网络尺寸而不是整个模型尺寸。

4.2　参数分析

通过实验来评估 FlexProtoFSSL 的阈值上限 τ 和辅助客户端选择对模型性能的影响。

4.2.1　阈值上限 τ 的影响

在 CIFAR-10 数据集上研究了 5 个不同的阈值 τ。如图 3 所示，τ 的最优选择约为 0.92，增大或减小都会导致性能衰减。算法 FlexProtoFSSL 在模型 ResNet9 上调整 τ 不仅影响阈值的上限，还影响该类估计的学习状态，因为它们是由落在 τ 之上的样本数量决定的。

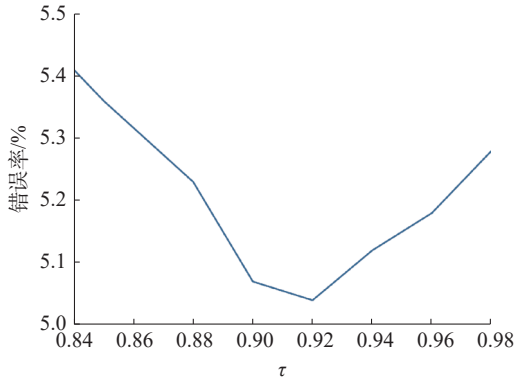


图 3　参数 τ 对模型性能的影响

Fig. 3　Effect of the parameter τ on model performance

4.2.2　辅助客户端选择的影响

由于算法 FlexProtoFSSL 仍然是基于原型来共享知识的，轻松地增加助手的数量并频繁地共享原型不会增加模型的计算和通信成本。为了分析该算法对可扩展性设计的影响，该算法在 CIFAR-10 数据集上对模型 ResNet9 讨论设置不同的辅助客户端数量 $|H_r|$ 参数对模型准确性的影响。

图 4 显示当 $|H_r|$=5 时模型的测试精度是最高的。如果使用更多的辅助客户端可能有助于性能改进，但轻量级设计对于在不给资源受限的客户端增加过多负担的情况下提高准确性非常重要。结果证明使用一定范围内不同数量的辅助客户端共享知识进行伪标记有助于解决梯度多样性。

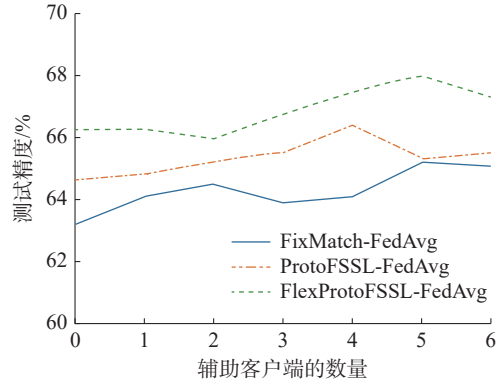


图 4　辅助客户端的数量对模型测试精度的影响

Fig. 4　Effect of the number of helper clients on the test accuracy of the model

4.2.3　温度控制参数的影响

为了评估温度参数对模型预测精度的影响，考虑在 IID 情况下使用 CIFAR-10 数据集，根据不同的温度参数 T 来评估模型的测试精度。选取了 4 个不同的 T 值来研究。表 4 给出了算法 FlexProtoFSSL 在模型 ResNet9 上 CIFAR-10 数据集的测试准确度的差异。当 T=0.1 时算法在模型上同时取得最好的测试精度，其次当 T=0.01 时模型精度略低，当 T=0.5 和 T=0.75 时由于温度较高，模型生成的类概率值没有较好地锐化导致模型泛化准确度变差。

表 4　参数 T 对算法性能的影响

Table 4　Algorithm performance effect of parameter T　%

T	FlexProtoFSSL-FedAvg	FlexProtoFSSL-FedProx
0.01	69.0	69.8
0.10	69.6	70.5
0.50	68.8	69.2
0.75	68.7	69.1

5　结束语

本研究针对数据异构分布的联邦半监督学习，重点是在不增加资源受限客户端的计算和通信负担的情况下提高模型精度。算法 FlexProtoFSSL 通过调整动态阈值从原型网络计算的类概率值中筛选出合格的来制作伪标签，并在客户端之间交换轻量级原型来强制客户端间一致性正则化。实验广泛地评估了 FlexProtoFSSL 的准确性、计算和通信成本、标签数据量、辅助客户端选择等。在最近的 FSSL 方法尝试中，观察到所提出的 FlexProtoFSS 实现了最先进的性能并且收敛速度不亚于其他算法。该方法为处理联邦半监督学习中的数据异质性和标签稀缺问题提供了一种创新的解决方案。

参考文献：

[1] SOHN K, BERTHELOT D, LI Chunliang, et al. FixMatch: simplifying semi-supervised learning with consistency and confidence[EB/OL]. (2020–01–21)[2023–11–13]. http://arxiv.org/abs/2001.07685.

[2] MALAVIYA S, SHUKLA M, KORAT P, et al. FedFAME: a data augmentation free framework based on model contrastive learning for federated semi-supervised learning[C]//Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. Tallinn: ACM, 2023: 1114–1121.

[3] FENG Siwei, LI Boyang, YU Han, et al. Semi-supervised federated heterogeneous transfer learning[J]. Knowledge-based systems, 2022, 252: 109384.

[4] PEI Xinjun, DENG Xiaoheng, TIAN Shengwei, et al. A knowledge transfer-based semi-supervised federated learning for IoT malware detection[J]. IEEE transactions on dependable and secure computing, 2023, 20(3): 2127–2143.

[5] WEN Tingjie, ZHAO Shengjie, ZHANG Rongqing. Federated semi-supervised learning through a combination of self and cross model ensembling[C]//2022 International Joint Conference on Neural Networks. Padua: IEEE, 2022: 1–8.

[6] ITAHARA S, NISHIO T, KODA Y, et al. Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-IID private data[J]. IEEE transactions on mobile computing, 2023, 22(1): 191–205.

[7] KIM W, PARK K, SOHN K, et al. Federated semi-supervised learning with prototypical networks[EB/OL]. (2022–05–27)[2023–11–13]. http://arxiv.org/abs/2205.13921.

[8] GAO Liang, FU Huazhu, LI Li, et al. FedDC: federated learning with non-IID data via local drift decoupling and correction[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 10102–10111.

[9] 窦勇敢, 袁晓彤. 基于隐式随机梯度下降优化的联邦学习 [J]. 智能系统学报, 2022, 17(3): 488–495.
DOU Yonggan, YUAN Xiaotong. Federated learning with implicit stochastic gradient descent optimization[J]. CAAI transactions on intelligent systems, 2022, 17(3): 488–495.

[10] 谭作文, 张连福. 机器学习隐私保护研究综述 [J]. 软件学报, 2020, 31(7): 2127–2156.
TAN Zuowen, ZHANG Lianfu. Survey on privacy preserving techniques for machine learning[J]. Journal of software, 2020, 31(7): 2127–2156.

[11] JEONG W, YOON J, YANG E, et al. Federated semi-supervised learning with inter-client consistency[C]//International Conference on Learning Representations. Virtual Only: ICLR, 2021: 901–914.

[12] WU Yaqiang, LI Yifei, ZHAO Tianzhe, et al. Improved prototypical network for active few-shot learning[J]. Pattern recognition letters, 2023, 172(C): 188–194.

[13] WANG Ruiqi, ZHANG Xuyao, LIU Chenglin. Meta-prototypical learning for domain-agnostic few-shot recognition[J]. IEEE transactions on neural networks and learning systems, 2022, 33(11): 6990–6996.

[14] SEHANOBISH A, KANNAN K, ABRAHAM N, et al. Meta-learning pathologies from radiology reports using variance aware prototypical networks[C]//Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track. Abu Dhabi: Association for Computational Linguistics, 2022: 332–347.

[15] XU Xin, DU Junping, XUE Zhe. Multi-level self-adaptive prototypical networks for few-shot node classification on attributed networks[J]. Neural computing and applications, 2023, 35(12): 9131–9144.

[16] PHAPHUANGWITTAYAKUL A, YING Fangli, GUO Yi, et al. Adaptive adversarial prototyping network for few-shot prototypical translation[J]. Journal of visual communication and image representation, 2023, 94: 103845.

[17] TAN Yue, LONG Guodong, LIU Lu, et al. FedProto: federated prototype learning across heterogeneous clients[J]. Proceedings of the AAAI conference on artificial intelligence, 2022, 36(8): 8432–8440.

[18] BERTHELOT D, CARLINI N, GOODFELLOW I, et al. MixMatch: a holistic approach to semi-supervised learning[EB/OL]. (2019–05–06)[2023–11–13]. http://arxiv.org/abs/1905.02249.

[19] LI Xiaorun, CAO Zeyu, ZHAO Liaoying, et al. ALPN: active-learning-based prototypical network for few-shot hyperspectral imagery classification[J]. IEEE geoscience and remote sensing letters, 2022, 19: 5508305.

[20] CASCANTE-BONILLA P, TAN Fuwen, QI Yanjun, et al. Curriculum labeling: revisiting pseudo-labeling for semi-supervised learning[J]. Proceedings of the AAAI conference on artificial intelligence, 2021, 35(8): 6912–6920.

[21] BERTHELOT D, CARLINI N, CUBUK E D, et al. ReMixMatch: semi-supervised learning with distribution alignment and augmentation anchoring[EB/OL]. (2019–11–21) [2023–11–13]. http://arxiv.org/abs/1911.09785.

[22] HAN Yue, LIU Yuhong, JIN Zhigang. Sentiment analysis via semi-supervised learning: a model based on dynamic threshold and multi-classifiers[J]. Neural computing and applications, 2020, 32(9): 5117–5129.

[23] KARIM N, MITHUN N C, RAJVANSHI A, et al. C-SFDA: a curriculum learning aided self-training framework for efficient source free domain adaptation[C]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Vancouver: IEEE, 2023: 24120–24131.

[24] ZHANG Bowen, WANG Yidong, HOU Wenxin, et al. FlexMatch: boosting semi-supervised learning with curriculum pseudo labeling[EB/OL]. (2021–10–15) [2023–11–13]. http://arxiv.org/abs/2110.08263.

[25] ARAZO E, ORTEGO D, ALBERT P, et al. Pseudo-labeling and confirmation bias in deep semi-supervised learning[C]//2020 International Joint Conference on Neural Networks. Glasgow: IEEE, 2020: 1–8.

[26] 王耀力, 刘晓慧, 李斌, 等. 流形嵌入的选择性伪标记与小样本数据迁移 [J]. 西北工业大学学报, 2021, 39(5): 1122–1129.
WANG Yaoli, LIU Xiaohui, LI Bin, et al. The manifold embedded selective pseudo-labeling algorithm and transfer learning of small sample dataset[J]. Journal of northwestern polytechnical university, 2021, 39(5): 1122–1129.

[27] 张英俊, 李牛牛, 谢斌红, 等. 课程学习指导下的半监督目标检测框架 [J/OL]. 计算机应用, (2023–12–20) [2024–04–26]. http://www.joca.cn/CN/10.11772/j.issn.1001-9081.2023081062.
ZHANG Yingjun, LI Niuniu, XIE Binhong, et al. Semi-supervised object detection framework guided by curriculum learning [J/OL]. Journal of computer applications, (2023–12–20)[2024–04–26]. http://www.joca.cn/CN/10.11772/j.issn.1001-9081.2023081062.

[28] XIE Qizhe, DAI Zihang, HOVY E, et al. Unsupervised data augmentation for consistency training[C]//Proceedings of the 34th International Conference on Neural Information Processing Systems. Vancouver: ACM, 2020:

6256–6268.

[29] FRANTAR E, ALISTARH D. SPDY: accurate pruning with speedup guarantees[EB/OL]. (2022–01–31)[2023–11–13]. http://arxiv.org/abs/2201.13096.

[30] LIU Ren, BIAN Fengmiao, ZHANG Xiaoqun. Binary quantized network training with sharpness-aware minimization[J]. Journal of scientific computing, 2022, 94(1): 16.

[31] MCMAHAN H B, MOORE E, RAMAGE D, et al. Communication-efficient learning of deep networks from decentralized data[EB/OL]. (2016–02–17)[2023–11–13]. http://arxiv.org/abs/1602.05629.

[32] LI Tian, SAHU A K, ZAHEER M, et al. Federated optimization in heterogeneous networks[EB/OL]. (2018–12–14)[2023–11–13]. http://arxiv.org/abs/1812.06127.

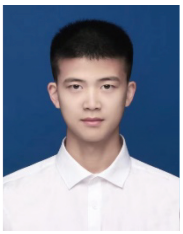
[33] LIANG Xiaoxiao, LIN Yiqun, FU Huazhu, et al. RSCFed: random sampling consensus federated semi-supervised learning[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 10144–10153.

[34] ZHANG Zhengming, YANG Yaoqing, YAO Zhewei, et al. Improving semi-supervised federated learning by reducing the gradient diversity of models[C]//2021 IEEE International Conference on Big Data. Orlando: IEEE, 2021: 1214–1225.

[35] COATES A, NG A, LEE H. An analysis of single-layer networks in unsupervised feature learning[C]//Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings. Fort Lauderdale: PMLR, 2011: 215–223.

[36] 杨寒雨, 赵晓永, 王磊. 数据归一化方法综述 [J]. 计算机工程与应用, 2023, 59(3): 13–22.
YANG Hanyu, ZHAO Xiaoyong, WANG Lei. Review of data normalization methods[J]. Computer engineering and applications, 2023, 59(3): 13–22.

作者简介:



陈涛, 硕士研究生, 主要研究方向为分布式机器学习、联邦学习。E-mail: 1033296297@qq.com。

谢在鹏, 副教授, 博士, 主要研究方向为分布式机器学习, 可持续计算理论及应用。获发明专利授权 15 项, 发表学术论文 30 余篇。E-mail: zaipengxie@hhu.edu.cn。

屈志昊, 副教授, 博士, 主要研究方向为边缘计算、边缘智能、联邦学习。主持国家自然科学基金青年基金、江苏省青年基金等项目 5 项。发表学术论文 20 余篇。E-mail: quzhihao@hhu.edu.cn。