arXiv:2303.02668v1 [cs.LG] 5 Mar 2023

面向物联网异构轻量级客户端聚合的知识增强半监督联邦学习

Jiaqi Wang* Shenglai Zeng† Zewei Long ‡ Yaqing Wang§ Houping Xiao¶ Fenglong Ma‖

摘要

联邦学习（FL）使多个客户端能够在不共享本地数据的情况下协同训练模型，并已在包括物联网（IoT）在内的多个领域取得了令人瞩目的成果。然而，物联网终端设备无法自动标注其收集的数据，导致客户端侧出现标签短缺问题。为了协同训练联邦学习模型，我们只能使用存储在服务器上的少量带标签数据。这是联邦学习中一种新颖但实用的场景，即服务器端标签半监督联邦学习（SemiFL）。尽管近期已有多种 SemiFL 方法被提出，但它们在模型设计中都未能关注个性化问题。物联网环境使 SemiFL 面临更大挑战，因为我们需要同时考虑设备的计算限制和通信成本。为应对这些新挑战，我们提出了一种名为 pFedKnow 的新型半监督联邦学习框架。pFedKnow 通过神经网络剪枝技术生成轻量级个性化客户端模型，以降低通信成本。此外，该框架将预训练大模型作为先验知识，用于指导个性化客户端模型的聚合，进一步提升框架性能。在图像和文本数据集上的实验结果表明，所提出的 pFedKnow 优于最先进的基线方法，同时显著降低了通信成本。所提出的 pFedKnow 源代码可在
h
t
t
ps://github.co
m/JackqqWang/pfedknow/tree/master 获取。

1 引言

联邦学习（FL）旨在中央服务器的协调下训练共享全局模型，同时将用户数据保留在本地客户端上，以保护用户数据隐私 [18]。这一新型学习范式已广泛应用于物联网（IoT）领域 [10, 21, 23, 34]。然而，目前大多数物联网领域的研究都假设本地设备存储的数据均已标注。这一假设并不切实际，因为用户通常缺乏标注生成数据的动力或专业知识。因此，在实际物联网应用中，使用未标注的客户端/用户数据进行训练似乎更为可行。为了提供更好的服务、吸引更多用户并进一步增加利润，企业可能愿意标注少量数据并将其存放在服务器上，以帮助训练出更准确的联邦学习模型。这催生了联邦学习中的一种新场景，即服务器端标签半监督联邦学习（半监督联邦学习）。

近期，已有一些半监督联邦学习方法被提出，例如 FedMatch [8], SSFL [39], FedMix [38],FedSEAL [1], 半监督联邦学习 [4], 和 FedTriNet [2]。这些模型主要致力于开发优质的全局模型，并将同一个全局模型提供给所有客户端。然而，由于物联网设备可能存储异构数据，通用的全局模型可能不足以体现每位物联网用户的独特性。因此，模型定制已成为物联网应用的迫切需求。此外，现有方法并非针对物联网应用而开发，也未考虑物联网设备的限制，例如计算资源有限和网络带宽受限。因此，亟需在物联网中开发一种新的半监督联邦学习范式。

然而，开发此类方法将面临若干挑战。首先，在标签位于服务器端的设定中不存在带标签数据，因此无法直接应用现有的个性化联邦学习技术，例如 pFedMe [28], , 因为这些技术都需要带标签的客户端数据。其次，为了节省计算资源并降低通信成本，物联网应用通常使用参数较少的模型

*宾夕法尼亚州立大学。电子邮箱：jqwang@psu.edu。前两位作者对本研究的贡献相同。
†电子科技大学。电子邮箱：shenglaizeng@gmail.com
‡伊利诺伊大学厄巴纳-香槟分校。电子邮箱：zeweil2@illinois.edu§普渡大学。电子邮箱：wang5075@purdue.edu
¶佐治亚州立大学。电子邮箱：hxiao@gsu.edu
‖宾夕法尼亚州立大学。电子邮箱：fenglong@psu.edu

版权 © 2023 SIAM。未经授权，禁止复制本文。

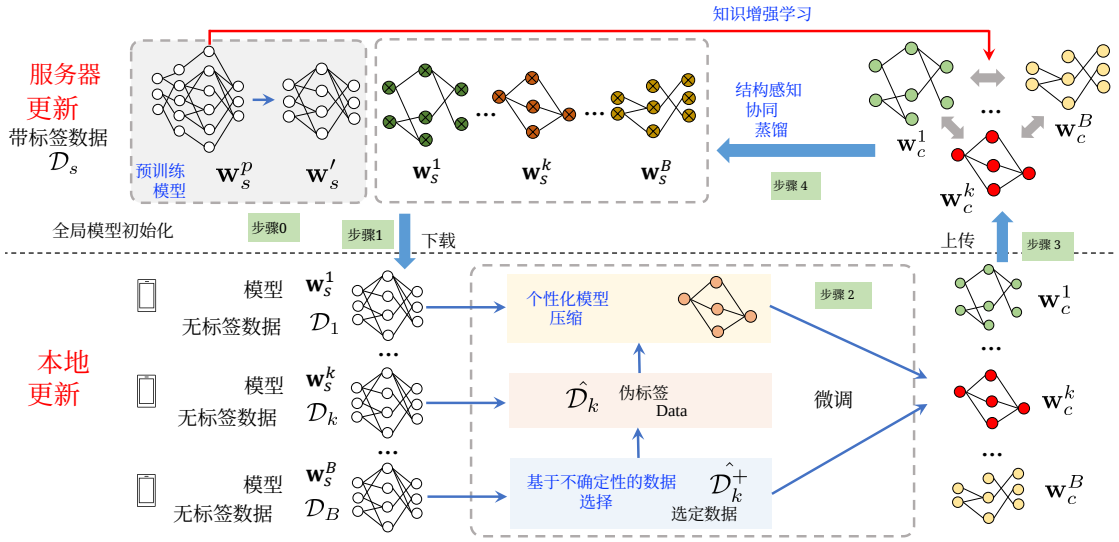


图1：所提出的 pFedKnow 模型概览，该模型包括初始化、本地个性化模型更新，以及通过结构感知协同蒸馏和知识增强学习进行的服务器更新。

，即 轻量级 模型。这些模型难以保持与大型模型相当的性能。因此，新的挑战在于如何保证新半监督联邦学习模型的性能，并同时实现客户端模型的个性化。

为应对上述挑战，本文针对物联网中标签位于服务器端的场景，提出了一种名为 pFedKnow 的新型半监督联邦学习框架。具体而言，我们提出将先进的 网络结构压缩 技术 [15, 32] 引入我们设计的 pFedKnow，以应对前文所述的挑战。这一设计的动机在于，每个客户端模型训练得到的网络参数都依赖于数据，这为通过定制不同的模型结构来实现不同客户端的个性化提供了一种新途径。这一设计有两方面的优势：（1）网络结构压缩主要取决于存储在各客户端的数据的独特特征，从而能够生成个性化客户端模型。由于各客户端存储的数据不同，即使使用相同的初始化轻量级模型，各客户端压缩后模型的网络结构也会有所不同。这样，我们就能为每个客户端获得个性化模型。（2）压缩轻量级模型还能进一步降低客户端与服务器之间的通信成本。

网络结构压缩是一把双刃剑，会显著增加服务器端聚合异构结构客户端模型的难度。为解决这一问题，我们提出了一种新颖的结构感知协同蒸馏机制，用于生成与相应客户端模型具有相同结构的聚合的个性化模型。我们

提出为服务器配备一个强大的预训练大模型作为先验知识，以进一步提升我们所设计的 pFedKnow 的性能。具体而言，该预训练模型不仅用于利用知识蒸馏技术初始化轻量级全局模型，还参与并指导服务器端个性化客户端模型的聚合。

图1展示了所提出的框架 pFedKnow 的概览，该框架包含两个模块，即本地更新和服务器更新，共分五个步骤。在步骤 0◯，我们首先在服务器上借助预训练模型 $\mathbf{w}_s^p$，使用带标签数据 $\mathcal{D}_s$ 训练并初始化一个轻量级全局模型$\mathbf{w}'_s$。该轻量级全局模型将分发给 B 个选定的客户端（即步骤 1◯），其中 $B \ll K$ 和 K 分别表示客户端总数。每个客户端 k 将使用无标签数据 $\mathcal{D}_k$ 和从服务器下载的模型$\mathbf{w}_s^k$ 来更新本地模型 $\mathbf{w}_c^k$（即步骤 2◯）。请注意，在第一个通信轮次中，$\mathbf{w}_s^k = \mathbf{w}'_s$。为了在每个客户端 k上学习轻量级个性化本地模型，我们为所有本地模型提出了一种新颖的个性化方法，从而得到具有不同网络架构的 $\{\mathbf{w}_c^1, \cdots, \mathbf{w}_c^B\}$。这些个性化模型将上传至服务器进行聚合（即步骤 3◯）。不同于 FedAvg [18], 等现有工作，我们提出了一种结构感知协同蒸馏与增强学习方法，借助其他上传的客户端模型和原始大型模型，在服务器上获得个性化本地模型（即步骤4◯）。这些蒸馏模型将在后续轮次中分发给选定的客户端。pFedKnow 将从步骤 1◯ 到步骤 4◯ 迭代运行，直至收敛。

总而言之，本文的贡献如下：

版权 © 2023 SIAM。未经授权，禁止复制本文。

表 1：主要符号。

符号	定义与描述
$\mathcal{D}_s$	服务器上的带标签数据集
$\mathcal{D}_k$	第 k个客户端上的无标签数据集
$\hat{\mathcal{D}}_k^+$	客户端 k上的高质量伪标签数据集
K	客户端数量
B	活跃/选定的客户端数量
$\mathbf{w}_s^p$	预训练模型
$\mathbf{w}_s'$	初始化的全局模型
$\mathbf{w}_s^k$	服务器上第 k 个客户端的蒸馏模型参数
$\mathbf{w}_c^k$	第 k 个客户端的本地模型参数
$\boldsymbol{\alpha}_i^k$	无标签数据的 logit 输出$\mathbf{w}_s^k$
u_i^k	无标签数据的不确定性分数 x_i^k

- 据我们所知，我们首次在联邦学习中将轻量级模型用于预热，并利用网络剪枝技术进一步定制具有不同结构的压缩本地模型，从而解决了物联网中本地设备计算能力有限和网络带宽受限的挑战。

- 我们提出了一种新的聚合方法，将网络结构感知的协作式蒸馏与大模型知识增强学习相结合。借助其他模型，即使其结构不同，也能获得个性化模型，并从预训练大模型中提取通用知识。

- 我们在图像和文本数据集上进行了大量实验，以展示所提出的 pFedKnow框架相较于最先进基线方法的有效性和效率。

2 任务与符号

本文关注现实中的标签存储于服务器的半监督学习场景。在这种设置下，服务器存储一小组带标签的数据，记为 $\mathcal{D}_s = \{(x_i^s,\ y_i^s)\}_{i=1}^{N_s}$，其中，$x_i^s$ 表示数据样本，y_i^s 表示相应的标签，N_s 表示带标签数据的数量。假设有 K 个客户端，且每个客户端 k 仅获得大量无标签数据，记为 $\mathcal{D}_k = \{x_i^k\}_{i=1}^{N_k}$，其中，$N_k$ 表示存储在第 k 个客户端中的无标签数据数量，$N_k \gg N_s$。本文的**目标**是利用服务器上的预训练模型$\mathbf{w}_s^p$ 和带标签数据 $\mathcal{D}_s$，以及存储在所有客户端中的无标签数据 $[\mathcal{D}_1, \cdots, \mathcal{D}_K]$，联合训练个性化客户端模型 $[\mathbf{w}_c^1,\ \cdots、\mathbf{w}_c^K]$。我们在表 1 中总结了后续章节所使用的关键符号。

3 方法

3.1 初始化由于每个客户端中都没有存储带标签数据，我们需要从服务器端开始进行模型学习。具体来说，我们首先在服务器上使用带标签数据初始化共享全局模型 $\mathbf{w}_s'$，然后将其分发给每个客户端。一种朴素的解决方案是仅使用 $\mathcal{D}_s$直接训练$\mathbf{w}_s'$。然而，少量的带标签数据可能会导致欠拟合问题。为了解决这一问题，我们提出在服务器端配备一个大型预训练模型$\mathbf{w}_s^p$，通过以下知识蒸馏技术帮助初始化更好的轻量级模型$\mathbf{w}_s'$：

$$\mathcal{L}_{ini} = \mathrm{CE}\left(f(\mathbf{x}^s; \mathbf{w}_s', \mathbf{y}^s) + \mathrm{KL}(\boldsymbol{\alpha}_s^p, \boldsymbol{\alpha}_s'\right), \tag{3.1}$$

其中，CE 表示交叉熵损失，$f(\cdot;\ \cdot)$ 表示轻量级神经网络，$\mathbf{x}^s$ 表示数据特征，$\mathbf{y}^s$ 表示标签向量。KL 表示库尔贝克-莱布勒散度。$\boldsymbol{\alpha}_s^p$ 和 $\boldsymbol{\alpha}_s'$ 分别表示预训练模型$\mathbf{w}_s^p$ 和轻量级初始模型$\mathbf{w}_s'$输出的逻辑值。pFedKnow 是一个通用框架，可以处理不同类型的数据。因此，在实验中，我们将选择不同的预训练模型和轻量级客户端模型。

在第一轮训练中，学习得到的轻量级模型 $\mathbf{w}_s'$[1] 将被分发给所有本地客户端，用于本地模型学习。从第二轮开始，每个本地模型都将接收更新后的个性化模型 $\mathbf{w}_s^k$，具体内容将在第 3.3 节中讨论。在不失一般性的情况下，以下各节中我们将使用$\mathbf{w}_s^k$ 作为服务器为每个客户端 k 分发的本地模型。

3.2 模型个性化（本地更新）遵循通用联邦学习模型的训练流程，我们将随机选择一小部分客户端，其数量为 B，以进行本地模型学习。假设第 k 个客户端被选中，则将使用无标签数据 $\mathcal{D}_k$ 进行其本地模型更新。

3.2.1 个性化模型压缩由于每个客户端中存储的只有无标签数据，传统的监督学习技术 [5, 11, 12, 18, 26]将无法奏效。因此，有必要设计一种新的学习范式，将无标签数据融入模型学习中。一种直观的方法是根据从服务器下载的模型 $\mathbf{w}_s^k$为无标签数据生成伪标签。随后，这些伪标记数据将用于本地模型训练。令 $\hat{}y_i^k$ 表示$\mathbf{w}_s^k$为每个无标签数据 x_i^k预测的伪标签，即 $\hat{}y_i^k = f(x_i^k; \mathbf{w}_s^k)$，并令 $\hat{}\ \mathcal{D}_k = \{(x_i^k, \hat{}y_i^k)\}_{i=1}^{N_k}$ 表示伪标注数据集。

为了降低客户端与服务器之间的通信成本，一个简单的方法是减少参数数量

[1]本文中，模型参数集 $\mathbf{w}$ 也用于表示模型架构。

版权 © 2023 SIAM 未经授权，禁止复制本文

，即学习轻量级本地模型。模型压缩是学习性能相当的轻量级模型的一种最先进方法，常用于计算机视觉 [6,7,31,33]和自然语言处理 [9, 16, 32, 35]领域。换言之，轻量级模型应具备与原始较大模型相似的预测能力。这促使我们学习轻量级$\mathbf{w}_c^k$，即使使用$\hat{}$伪标注数据 $\mathcal{D}_k$ 也如此。

因此，在每个客户端 k 上，我们直接应用现有的**网络剪枝**技术 [15, 32]，基于服务器下载的模型 $\mathbf{w}_s^k$ 和伪标注数据 $\hat{}\,\mathcal{D}_k$，学习轻量级模型 $\mathbf{w}_c^k$，即:

$$\mathbf{w}_c^k = g(\hat{\mathcal{D}}_k; \mathbf{w}_s^k, \gamma), \tag{3.2}$$

其中，$g(;,)$ 是压缩函数，γ 是用于控制压缩率的超参数。对于图像数据，我们采用网络瘦身方法 [15]；对于文本数据，则采用 SparseBERT [32] 中的权重剪枝方法。需要注意的是，轻量级模型 $\mathbf{w}_c^k$ 保留了$\mathbf{w}_s^k$ 的关键网络架构，但参数更少。

由于不同本地客户端上的伪标注数据 $\hat{}\,\mathcal{D}_k(k = 1, \cdots, B)$ 不同，即使使用相同的模型压缩技术 $g(\cdot;\cdot)$，我们也无法保证压缩后的轻量级模型 $\{\mathbf{w}_c^1、\cdots、\mathbf{w}_c^B\}$ 具有相同的网络架构。这一独特特性使我们能够在每个客户端上开展个性化半监督联邦学习。

直接将$\mathbf{w}_c^k$ 上传到服务器，必然会降低通信成本。然而，当前的本地模型$\mathbf{w}_c^k$仅在伪标注数据$\hat{}\,\mathcal{D}_k$上模仿$\mathbf{w}_s^k$的预测行为，并未真正借助本地数据更新模型参数。遗憾的是，由于$\hat{}y_i^k$的伪标签质量可能差异很大，我们无法使用$\mathbf{w}_s^k$ 中的全部数据来更新$\mathbf{w}_c^k$。为了解决这一问题，我们提出使用不确定性分数筛除质量较低的伪标签。不确定性分数越小，伪标签质量越高。

3.2.2 基于不确定性的数据选择令$\boldsymbol{\alpha}_i^k \in \mathbb{R}^{|\mathcal{C}|}$ 表示在无标签数据 x_i^k上使用$\mathbf{w}_s^k$ 得到的 logit，其中 $|\mathcal{C}|$ 为标签类别的数量。对于不同类型的数据，$\mathbf{w}_s^k$ 具有不同的网络架构。因此，我们将采用不同的方法生成不确定性分数，但这些方法都基于学习得到的 logit $\boldsymbol{\alpha}_i^k$。

对于图像数据，我们采用先前工作 [24] 提出的方法来量化不确定性分数，如下所示:

$$u_i^k = \frac{|\mathcal{C}|}{\sum_{c=1}^{|\mathcal{C}|}(\mathrm{ReLU}(\alpha_{i,c}^k) + 1)}. \tag{3.3}$$

对于文本数据，我们直接使用通过 $\boldsymbol{\alpha}_i^k$在 Softmax层上学习得到的每个无标签数据 x_i^k的概率分布 $[p_{i,1}^k, \cdots, p_{i,|\mathcal{C}|}^k]$。然后，不确定性分数建模如下:

$$u_i^k = 1 - \max\{p_{i,1}^k, \cdots, p_{i,|\mathcal{C}|}^k\}. \tag{3.4}$$

如果 $u_i^k \le \delta$，则将 $(x_i^k, \hat{}y_i^k)$ 选用于本地模型更新，其中 δ 是阈值超参数。为避免歧义，我们用 $(x_i^k+, \hat{}\,y_i^k+)$ 表示选定数据。这种基于不确定性的数据选择方法使我们能够在每个客户端 k 上获得高质量的伪标签数据集 $\hat{}\mathcal{D}_k^+ = \{(x_i^k+, \hat{}\,y_i^k+)\}_{i=1}^{N_k^+} \subseteq \hat{}\mathcal{D}_k$，其中 N_k^+ ($\le N_k$) 是选定数据的数量。

3.2.3 个性化模型更新 基于选定的高质量伪数据 $\hat{}\mathcal{D}_k^+$，我们可以使用以下损失函数对轻量级模型 $\mathbf{w}_c^k$ 进行微调:

$$\mathcal{L}_k = \mathrm{CE}\left(f_k\left(\mathbf{x}^k+; \mathbf{w}_c^k\right), \hat{\mathbf{y}}^k+\right), \tag{3.5}$$

其中，$\mathbf{x}^k+$ 是数据特征，$\hat{}\,\mathbf{y}^k+$ 是相应的高质量伪标签向量。最后，这些经过微调的轻量级模型参数$\{\mathbf{w}_c^k, \cdots, \mathbf{w}_c^B\}$ 将上传至服务器。接下来，我们将在下一小节介绍如何进行模型融合。

3.3 协同蒸馏（服务器更新）服务器收到本地上传的模型$\{\mathbf{w}_c^1, \cdots, \mathbf{w}_c^B\}$后，如果采用 FedAvg [18]，等现有联邦学习方法，我们会聚合$\{\mathbf{w}_c^1, \cdots, \mathbf{w}_c^B\}$，生成共享全局模型，然后再次将其分发给每个本地端。然而，如第 3.2.1 节所述，轻量级模型的网络架构 $\{\mathbf{w}_c^1, \cdots, \mathbf{w}_c^B\}$ 可能彼此存在显著差异，导致现有模型聚合方法失效。因此，我们需要设计一种新的学习范式，以便在服务器上进行异构模型融合。

然而，设计这样一种新模型并非易事，因为它需要满足以下要求：首先，为了实现个性化学习，我们既需要保留当前模型$\mathbf{w}_c^k$的关键特征，也要考虑其他已上传的模型参数 $\{\ \ _c^j\}_{j=1}^{B-1}\ \ j \neq$ ^{k} $\mathbf{w}$ ( )，以吸收共通知识，从而进一步提升学习效果。因此，仅使用 $\mathcal{D}_s$ 对模型$\mathbf{w}_c^k$ 进行微调，并将其作为更新后的$\mathbf{w}_k^s$，并不是合适的解决方案。其次，联邦学习需要迭代训练每个本地模型。因此，如果模型架构频繁且大幅变化，模型训练将难以收敛。

版权 ©2023 SIAM。未经授权，禁止复制本文。

此外，由于每个本地模型都是原始模型$\mathbf{w}_s'$的子结构，这可能导致轻量级模型缺失原始预训练模型所学习到的通用知识。为避免这一问题，要求每个模型在模型融合过程中保持其原始架构，同时聚合存储在服务器端的大型模型$\mathbf{w}_s^p$中的通用知识。

为实现这三个目标，我们提出一种新颖的结构感知与知识增强的协同蒸馏 方法，用于学习个性化聚合模型$\mathbf{w}_s^k$，其网络架构与$\mathbf{w}_c^k$相同。此外，即使上传的模型来自网络架构各异的客户端，所提出的结构感知协同蒸馏方法仍可借助其他本地轻量级模型和预训练模型交替利用存储在服务器上的带标签数据 $\mathcal{D}_s$ 提升学习效果。

3.3.1 结构感知相似性学习具体而言，在学习模型$\mathbf{w}_s^k$时，我们将$\mathbf{w}_c^k$ 作为领导模型，并将 $\{\mathbf{w}_c^j\}_{j=1}^{B-1}$ 作为辅助模型，同时考虑它们的网络结构。直观而言，如果两个不同客户端上的数据分布相似，那么轻量级客户端模型的网络结构也会相似，进而产生相似的输出。因此，不同辅助模型的贡献各不相同，区分它们的重要性至关重要。基于这一点，我们提出一种新方法，为每个辅助模型自动分配权重，该权重由领导模型与辅助模型之间的结构相似度衡量。

为实现这一目标，我们首先通过比较$\mathbf{w}_c^k$ 和$\mathbf{w}_s'$，得到一个二进制掩码向量 $\boldsymbol{\theta}_k$ ，用于指示哪些权重被剪枝。同样，对于每个辅助模型，我们也可以得到这样的掩码向量$\boldsymbol{\theta}_j$ （$j \neq k$）。然后，我们可以计算 $\boldsymbol{\theta}_k$ 和 $\boldsymbol{\theta}_j$ 之间的相似度分数，即:

$$\beta_j^k = \cos(\boldsymbol{\theta}_k, \boldsymbol{\theta}_j), \tag{3.6}$$

其中，cos(·, ·)表示余弦相似度。由于我们有 $B-1$ 个辅助模型，因此可以得到 $B-1$ 个相似度分数。最后，我们对这些分数进行归一化，并令$\sum_{j=1}^{B-1} \beta_j^k = 1$。

3.3.2 知识增强的协同蒸馏 为了训练所设计的结构感知型协同蒸馏方法，我们遵循通用的知识蒸馏方法 [37]，结合分类损失（即交叉熵）以及加权后的辅助模型和预训练模型如下:

$$\begin{aligned}\mathcal{L}_s^k =& \mathrm{CE}\left(f_k(\mathbf{x}^s; \mathbf{w}_s^k), \mathbf{y}^s\right) + \mathrm{KL}(\boldsymbol{\alpha}_{avg}^k, \boldsymbol{\alpha}_s^k) \\ &+ \mathrm{KL}(\boldsymbol{\alpha}_s^p, \boldsymbol{\alpha}_s^k),\end{aligned} \tag{3.7}$$

其中， $\boldsymbol{\alpha}_s^k$ 是来自领导模型$\mathbf{w}_c^k$的软目标预测或逻辑值，$\boldsymbol{\alpha}_s^p$ 表示来自大型预训练模型的逻辑值， $\boldsymbol{\alpha}_{avg}^k$ 表示所有辅助模型的逻辑值的加权平均，其定义为 $\boldsymbol{\alpha}_{avg}^k = \sum_{j=1}^{B-1} \beta_j^k * \boldsymbol{\alpha}_s^j$.

请注意，在优化式（3.7）时，$\mathbf{w}_s^k$ 使用$\mathbf{w}_c^k$ 作为初始化。如果服务器的计算能力足够强，所提出的结构感知协同蒸馏方法可以并行更新所有模型 $\{\mathbf{w}_s^1, \cdots, \mathbf{w}_s^B\}$。因此，这种设计不会对实际应用中的计算效率造成太大影响。

在服务器上训练好每个个性化模型 $\mathbf{w}_s^k$ 后，这些模型随后将分发给相应的本地客户端。每个本地客户端将再次更新并上传参数。这一迭代过程将持续进行，直至模型收敛或达到最大迭代次数。

4 实验

本节中，我们在不同类型的数据集上开展了一系列实验，以验证所提出框架的有效性。我们旨在回答以下研究问题:

- **RQ1**：在不同类型的数据上，我们提出的框架的性能能否与最先进的半监督联邦学习算法相媲美，甚至优于它们?

- **RQ2**：在保持准确率表现相当的情况下，在迭代通信机制下，我们的框架能节省多少通信成本?

- **RQ3**：在半监督设置下，各模块如何发挥作用，使框架实现高效的个性化联邦学习?

4.1 实验设置 为评估我们所提出框架的泛化能力，我们在两类数据上对其进行了验证。**(1) 图像数据集**。我们分别在独立同分布和非独立同分布数据分布设置下，使用 SVHN 和 CIFAR-10 对图像分类开展实验。**(2) 文本数据集**。我们使用 AG新闻数据集验证文本分类任务的性能。

本文聚焦于半监督联邦学习设置，因此我们选择 FedMatch [8],

版权 © 2023 SIAM 未经授权，禁止复制本文

表 2：与半监督联邦学习基线方法的图像分类性能比较。

数据集	SVHN		CIFAR-10	
设置	IID	非独立同分布	IID	非独立同分布
FedMatch [8]	78.34%	74.76%	64.70%	61.12%
SSFL [39]	76.06%	70.29%	64.45%	60.33%
FedMix [38]	78.45%	71.76%	63.68%	61.79%
FedSEAL [1]	72.64%	69.02%	62.39%	60.07%
半监督联邦学习 [4]	84.65%	82.15%	70.79%	68.66%
pFedKnow	**85.31%**	**84.79%**	**71.05%**	**69.81%**

将 SSFL [39], FedMix [38], FedSEAL [1], 和半监督联邦学习 [4] 作为基线方法。

对于图像数据，我们使用 VGG19 [25] 作为预训练模型。对于文本数据，我们使用 BERT [3] 作为大型模型。所选的轻量级模型包括：用于图像数据的卷积神经网络（CNN），由六个卷积层和三个线性层组成（3,745,520 个参数）；以及用于文本数据的轻量级 BERT 模型（71,766,890 个参数）。请注意，我们从头开始训练卷积神经网络和轻量级 *BERT* 模型。

4.2 图像数据集上的评估

4.2.1 性能评估为回答 **RQ1**，我们选择本地模型文本准确率的平均值作为评估指标。我们对每种方法运行三次，并在表 2 中报告平均准确率 。对于图像分类任务，我们得到以下观察和分析。在**独立同分布设置**下，我们的方法在 SVHN 和 CIFAR-10 数据集上的测试准确率分别达到最高的 85.31% 和 71.05%。然而，FedMatch、FedMix 和 SemiFL 都采用了数据增强技术，实际上增加了训练数据的数量，从而进一步提升了性能。与基线方法相比，pFedKnow 不会添加任何其他合成数据。尽管模型结构有所缩减且参数更少，协同蒸馏过程仍使每个本地模型能够从其他活跃的本地模型中提取通用知识。

在**非独立同分布设置**下，我们提出的框架在 SVHN 和 CIFAR-10 数据集上的表现优于最先进的半监督联邦学习基线方法。这表明，与其他半监督联邦学习方法相比，我们的个性化机制能够更好地应对数据异质性挑战，因为每个本地模型都会根据本地数据，通过模型结构压缩和进一步的参数微调进行定制。

4.2.2 通信效率分析为回答 **RQ2**，我们对通信成本进行了分析

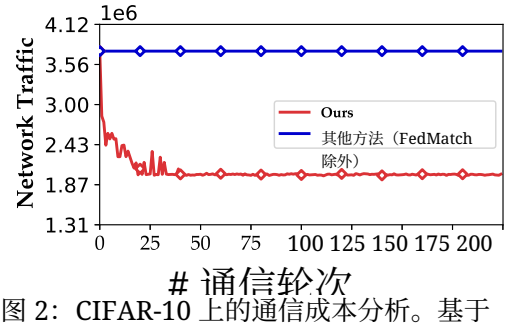


图 2：CIFAR-10 上的通信成本分析。基于 CIFAR-10 数据集。结果见图 2。 x 轴表示通信轮数，y 轴表示服务器与活跃客户端之间的通信量，即上传的模型参数数量。

蓝色线条表示除 FedMatch 外其他基线方法的网络流量。尽管 FedSEAL [1], 中存在一个从服务器传输到客户端的向量 τ_t ，但与模型参数大小和总流量相比，该数值可以忽略不计。我们通过计算（1 - 我们的通信成本/常规通信成本）$\times$ 100%，报告相较于初始轻量级模型的通信成本节省比例。平均而言，pFedKnow 与基线方法相比可节省 44.9% 的通信成本。

对于**FedMatch** [8], ，它通过参数分解降低通信成本，并传输差值 $\Delta\psi$ 和 $\Delta\sigma$。在其“标签位于服务器端”场景中，当活跃客户端比例为 0.05 时，我们计算出的平均通信成本为 33.50%（$\mathrm{F} = 0.05$）。将活跃客户端比例从 0.1 调整至 0.05，以匹配其设置后，pFedKnow 的平均通信成本仅为 22.45%，相当于 FedMatch 通信成本的 67.01%。

4.2.3 消融实验为回答**RQ3**，我们在图像数据集上针对 pFedKnow 框架开展消融实验，以探究知识增强、本地压缩和协同蒸馏这几个关键模块的作用。我们采用以下四种精简或修改后的模型作为基线方法。前三种模型未使用预训练模型来增强学习，最后一种用于测试本地客户端压缩的有效性。

AS-1：不使用本地模型压缩、预训练大模型和协同蒸馏，服务器通过 FedAvg 聚合本地模型，然后对全局模型进行微调。AS-1 可视为$Semi\text{-}FedAvg$。

AS-2：不使用所设计的协同蒸馏和预训练大模型，我们仅通过填充剪枝权重来采用基本的平均方法

版权 © 2023 SIAM 未经授权，禁止转载本文

表3：图像数据集上的消融实验。

数据集	SVHN		CIFAR-10	
设置	IID	非独立同分布	IID	非独立同分布
AS-1	72.50%	67.60%	63.70%	61.56%
AS-2	67.56%	65.04%	59.74%	59.05%
AS-3	79.55%	77.52%	63.67%	63.47%
AS-4	84.62%	80.16%	67.05%	65.71%
pFedKnow	**85.31%**	**84.79%**	**71.05%**	**69.81%**

采用 0，但保留本地模型压缩模块。

AS-3：不使用预训练大模型进行知识增强学习，我们仅使用轻量级模型执行系统更新，同时保留 pFedKnow 的所有其他模块。

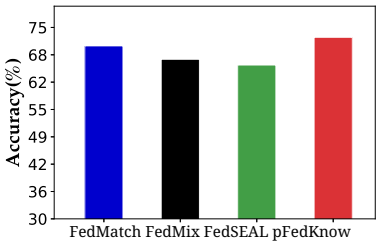
AS-4：不使用本地模型压缩，其他操作与我们提出的方法相同。

表3展示了实验结果。我们可以观察到，在 pFedKnow 中减少一个或多个关键组件会导致性能显著下降。与 AS-1 相比的结果清楚地表明，基本的半监督模型聚合不足以解决具有挑战性的半监督联邦学习问题。在 AS-2 中，我们采用客户端压缩技术生成个性化本地模型，但使用朴素方法聚合全局模型。这种简单的方法会在模型聚合过程中引入额外误差，进而导致其性能在所有基线方法中最差。AS-3 的性能远优于 AS-1 和 AS-2。在 AS-3 中，我们采用了设计的协同蒸馏，这是一种获取个性化客户端模型的有效方法。这些结果清楚地证实了所设计的协同蒸馏具有优越性。AS-4 将预训练模型用作先验知识，其性能优于其他三种基线方法，这证明在联邦学习中引入预训练模型是有益的。此外，与 pFedKnow 相比，AS-4 的性能有所下降。这一结果表明，在模型设计中采用客户端压缩是必要的。

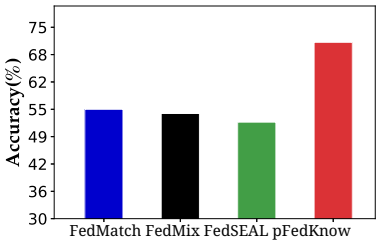
4.3 文本数据集上的评估 为了验证所提出的 pFedKnow 的泛化能力，我们还在 AG 新闻数据集上进行了实验。

4.3.1 性能评估 由于自然语言处理领域关于联邦学习的研究有限 [14] ，我们选择了与图像任务相同的基线方法。然而，根据针对图像任务的直觉和典型设计，SSFL [39]和 FedMix [38] 并不适用于文本任务。SSFL 的做法是用组归一化替换批归一化。在类似 BERT 的模型中，采用的是层归一化，而不是

而不是批归一化。在 FedMix 中，该操作



（a）独立同分布设置



（b）非独立同分布设置

图3：AG 上的性能比较。

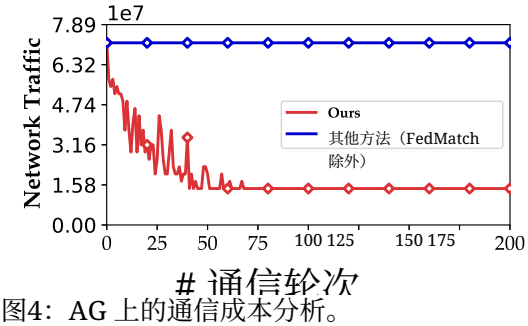


图4：AG 上的通信成本分析。

应用于图像对文本任务没有意义。因此，我们删除这两个基线方法，以确保公平比较。由于基线方法需要以增强数据作为输入，我们遵循文献 [30] 进行数据增强。结果如图3所示。我们可以观察到与图像数据集上的实验相似的结果。值得强调的是，与最佳基线方法 FedMatch 相比，pFedKnow 在非独立同分布设置下显著提升了性能。这些结果再次证明了 pFedKnow 的有效性。

4.3.2 通信效率分析我们还分析了文本数据上的通信成本。结果如图4所示。具体而言，轻量级 BERT 模型总共进行了四次压缩，使平均通信成本降低了73.6%。

5 相关工作

5.1 物联网中的联邦学习 联邦学习已广泛应用于物联网。然而，物联网环境中的有限带宽对联邦学习的通信效率提出了严峻要求。

版权 © 2023 SIAM 未经授权，禁止复制本文

[22] 使用稀疏化和最优 Golomb 编码等压缩技术，减少客户端与服务器之间的流量。 [20]使用一种新颖的 Count Sketch 数据结构来压缩上传的梯度。CMFL [17] 通过避免上传与全局更新不太一致的本地更新，减少了参数交换。 [27] 尝试通过使用梯度校正和批归一化来加速模型收敛并减少通信轮次，从而使优化器能够妥善处理累积的不重要梯度。

物联网中联邦学习的应用也受到数据异质性的影响，而数据异质性可能由传感器类型、环境、任务等因素引起。一些研究人员已致力于缓解这一问题。例如， [36] 采用高效的分组方法，将异质客户端分配到同质组中，并采用新颖的顺序到并行训练模式，以协调组内及组间的训练。 [29] 采用数据增强、混合训练损失和系统级训练策略来解决数据不平衡问题。 [19]构建了一个基于强化学习的智能中央服务器，用于捕捉具有偏倚数据的客户端之间的复杂模式，并最终制定具有准最优性能的分配方案。 [13] 对不同工厂中相似的用户进行下采样，以构建参与联邦学习训练的同质超级节点，从而有效抵抗数据异质性。

5.2 半监督联邦学习与传统监督式联邦学习相比， [12, 18], 半监督联邦学习更加实用，也更具挑战性。近两年来，已有多项研究提出将联邦学习与半监督学习技术相结合。在 [8], 中，研究人员提出了一种新的一致性损失，以最大化不同本地模型之间的一致性，并采用解耦学习从带标签数据和无标签数据中提取可靠知识。 [38, 39] 设计了基于分组和加权的聚合策略，以减弱数据异质性的负面影响。在 [1], 中，作者将自集成学习和互补负学习引入联邦学习。FedTriNet [2] 采用三个网络和动态质量控制机制，为无标签数据生成高质量伪标签，随后结合伪标签数据与真实标记数据重新训练模型。

尽管这些模型能够提升预测性能，但它们都没有考虑模型个性化。此外，只有少数模型在设计时考虑了通信成本。

本研究旨在同时解决这些具有挑战性的问题，并提出一种高效的个性化半监督联邦学习框架 pFedKnow。

6 结论

本文提出了一种轻量、高效且个性化的半监督联邦学习框架 pFedKnow，可广泛应用于物联网中的图像和文本分类任务。我们在图像和文本数据集上进行了实验，证明了 pFedKnow 在半监督设置下应对数据异质性和通信成本挑战的有效性。

参考文献

[1]Jieming Bian、Zhu Fu 和 Jie Xu。FedSEAL：结合自集成学习与负学习的半监督联邦学习。 *arXiv preprint arXiv:2110.07829* 2021 ，。

[2] Liwei Che、Zewei Long、Jiaqi Wang、Yaqing Wang、Houping Xiao 和 Fenglong Ma。FedTriNet：一种用于联邦半监督学习的三方伪标签方法。载于 *BigData*，第715–724页。IEEE，2021年。[3] Jacob Devlin、Ming-Wei Chang、Kenton Lee 和 Kristina Toutanova。BERT：用于语言理解的深度双向 Transformer 预训练。 *arXiv preprint arXiv:1810.04805* 2018 ，。

[4] Enmao Diao、Jie Ding 和 Vahid Tarokh。半监督联邦学习：面向无标签客户端的通信高效方法。 *NeurIPS*，2022年。[5] Chelsea Finn、Pieter Abbeel 和 Sergey Levine。用于快速适应深度网络的模型无关元学习。载于 *ICML*，第1126–1135页。PMLR，2017年。[6] Jonathan Frankle 和 Michael Carbin。彩票假设：寻找稀疏、可训练的神经网络。 *arXiv preprint arXiv:1803.03635*，2018年。[7] Yunchao Gong、Liu Liu、Ming Yang 和 Lubomir Bourdev。利用矢量量化压缩深度卷积网络。 *arXiv preprint arXiv:1412.6115* 2014 ，。

[8] Wonyong Jeong、Jaehong Yoon、Eunho Yang 和 Sung Ju Hwang。基于客户端间一致性的联邦半监督学习。 *arXiv e-prints*，第arXiv–2006页，2020年。[9] Xiaoqi Jiao、Yichun Yin、Lifeng Shang、Xin Jiang、Xiao Chen、Linlin Li、Fang Wang 和 Qun Liu。TinyBERT：面向自然语言理解的BERT蒸馏。*arXiv preprint arXiv:1909.10351*，2019年。[10] Latif U Khan、Walid Saad、Zhu Han、Ekram Hossain 和 Choong Seon Hong。面向互联

版权 © 2023 SIAM。未经授权，禁止转载本文。

网的联邦学习：最新进展、分类与开放挑战。 *IEEE Commun. Surv. Tutor.*，2021年。[11] Daliang Li 和 Junpu Wang。FedMD：通过模型蒸馏实现异构联邦学习。 *arXiv preprint arXiv:1910.03581* 2019 ，。[12] Tian Li、Anit Kumar Sahu、Manzil Zaheer、Maziar Sanjabi、Ameet Talwalkar 和 Virginia Smith。异构网络中的联邦优化。 *MLS*，2:429–450，2020年。[13]Zonghang Li、Yihong He、Hongfang Yu、Jiawen Kang、Xiaoping Li、Zenglin Xu 和 Dusit Niyato。工业物联网中通过分组客户端选择实现对数据异质性的鲁棒联邦学习。 *IEEE Internet of Things Journal* 2022 ，。[14] Bill Yuchen Lin、Chaoyang He、Zihang Zeng、Hulin Wang、Yufen Huang、Mahdi Soltanolkotabi、Xiang Ren 和 Salman Avestimehr。FedNLP：用于自然语言处理任务的联邦学习方法基准测试。 *arXiv preprint arXiv:2104.08815*，2021年。[15] Zhuang Liu、Jianguo Li、Zhiqiang Shen、Gao Huang、Shoumeng Yan 和 Changshui Zhang。通过网络瘦身学习高效卷积网络。载于 *ICCV*，第2736–2744页，2017年。[16] Ekaterina Lobacheva、Nadezhda Chirkova 和 Dmitry Vetrov。循环神经网络的贝叶斯稀疏化。 *arXiv preprint arXiv:1708.00077*，2017年。[17] Wang Luping、Wang Wei 和 Li Bo。CMFL：降低联邦学习的通信开销。载于*ICDCS*，第954–964页。IEEE，2019年。[18] Brendan McMahan、Eider Moore、Daniel Ramage、Seth Hampson 和 Blaise Aguera y Arcas。基于去中心化数据的深度网络通信高效学习。载于 *Artificial intelligence and statistics* 1273 1282 2017 ，第–页。 PMLR，。[19] Junjie Pang、Yan Huang、Zhenzhen Xie、Qilong Han 和 Zhipeng Cai。实现异构性：面向物联网的自组织联邦学习框架。 *IEEE Internet of Things Journal* 8 5 3088 3098, 2020 ，（）：–。 [20]Daniel Rothchild、Ashwinee Panda、Enayat Ullah、Nikita Ivkin、Ion Stoica、Vladimir Braverman、Joseph Gonzalez 和 Raman Arora。FetchSGD：利用草图实现通信高效的联邦学习。载于 *ICML*，第8253–8265页。PMLR，2020年。[21] Sumudu Samarakoon、Mehdi Bennis、Walid Saad 和 Merouane Debbah。面向超可靠低时延v 2 *GLOBE*- v通信的联 *COM* 1 7 2018 邦学习。载于，第–页。IEEE，。[22] Felix Sattler、Simon Wiedemann、Klaus-Robert M¨uller 和 Wojciech Samek。基于非独立同分布数据的鲁棒且通信高效的联邦学习。 *IEEE transactions on neural networks and learning systems*，31 9 3400 3413，2019（）：–。[23] Stefano Savazzi、Monica Nicoli 和 Vittorio Rampa。协作设备的联邦学习：面向大规模物联网网络的 *IEEE Internet* 共识方法。*of Things Journal* 7 5 4641 4654, 2020 ，（）：–。[24] Murat Sensoy、Lance Kaplan 和 Melih Kandemir。用于量化分类不确定性的证据深度学习 确定性。 *arXiv preprint arXiv:1806.01768*, 2018年。[25] Karen Simonyan 和 Andrew Zisserman。用于大规模图像识别的极深卷积网络。 *arXiv preprint arXiv:1409.1556*, 2014年。[26] Virginia Smith、Chao-Kai Chiang、Maziar Sanjabi 和 Ameet Talwalkar。联邦多任务学习。 *arXiv preprint arXiv:1705.10467*, 2017年。[27] Haifeng Sun、Shiqi Li、F Richard Yu、Qi Qi、Jingyu Wang 和 Jianxin Liao。面向物联网中借助边缘计算实现通信高效的联邦学习。 *IEEE Internet of Things Journal* 7 11 11053 11067, 2020 ，( ): – 。 [28] Canh T Dinh、Nguyen Tran 和 Josh Nguyen。基于 Moreau 包络的个性化联邦学习。 *Advances in Neural Information Processing Systems*, 33 21394 21405, 2020 : – 。[29] Jiaqi Wang、Cheng Qian、Suhan Cui、Lucas Glass 和 Fenglong Ma。面向新冠疫苗副作用预测的联邦学习。 *ECMLPKDD*, 2022年。[30] Jason Wei 和 Kai Zou。EDA：提升文本分类任务性能的简易数据增强技术。 *arXiv preprint arXiv:1901.11196*, 2019。[31] Bichen Wu、Forrest Iandola、Peter H Jin 和 Kurt Keutzer。Squeeze Det：用于自动驾驶实时目标检测的统一、小型、低功耗全卷积神经网络。载于 *CVPR workshops*，第129–137页，2017年。[32] Dongkuan Xu、Ian EH Yen、Jinxi Zhao 和 Zhibin Xiao。在预训练与微调范式下重新思考网络剪枝。 *arXiv preprint arXiv:2104.08682* 2021 ，。[33] Xiyu Yu、Tongliang Liu、Xinchao Wang 和 Dacheng Tao。通过低秩稀疏分解压缩深度模型。载于 *CVPR* 7370 7379, ，第–页。 2017 [34] Binhang Yuan、Song Ge 和 Wenhui Xing。面向医疗保健物联网设备的联邦学习框架。*arXiv preprint arXiv:2005.05083*, 2020年。[35] Ofir Zafrir、Guy Boudoukh、Peter Izsak 和 Moshe Wasserblat。Q 8 8 *EMC2*- BERT：*NIPS* 36 39 2019 量化比特 BERT。载于，第–页。IEEE，。[36] Shenglai Zeng、Zonghang Li、Hongfang Yu、Yihong He、Zenglin Xu、Dusit Niyato 和 Han Yu。通过分组顺序到并行训练实现异构联邦学习。载于 *DSFAA*，第455–471页。Springer，2022年。[37] Ying Zhang、Tao Xiang、Timothy M Hospedales 和 Huchuan Lu。深度互学习。载于 *CVPR*，第4320–4328页，2018年。[38] Zhe Zhang、Shiyao Ma、Jiangtian Nie、Yi Wu、Qiang Yan、Xiaoke Xu 和 Dusit Niyato。非独立同分布数据下的半监督联邦学习：算法与系统设计。 *arXiv preprint arXiv:2110.13388*, 2021年。[39] Zhengming Zhang、Yaoqing Yang、Zhewei Yao、Yujun Yan、Joseph E Gonzalez 和 Michael W Mahoney。通过降低模型的梯度多样性改进半监督联邦学习。 *arXiv preprint arXiv:2008.11364* 2020 ，。

版权 © 2023 SIAM。未经授权，禁止转载本文。