《中国通信》

数字孪生辅助的异构联邦学习知识蒸馏框架

Xiucheng Wang[1], Nan Cheng[1], *, Longfei Ma[1], Ruijin Sun[1],Rong Chai[2],Ning Lu[3]

[1]西安电子科技大学电信工程学院，中国西安 710071[2]重庆邮电大学重庆市移动通信技术重点实验室，中国重庆 400065[3]加拿大安大略省金斯顿女王大学电气与计算机工程系，K7L 3N6[*]通讯作者，电子邮箱：dr.nan.cheng@ieee.org

arXiv:2303.06155v1 [cs.LG] 10 Mar 2023

摘要： 本文针对联邦学习（FL）系统中的异质性，提出了一种由知识蒸馏（KD）驱动的联邦学习训练框架。在该框架中，每位用户都可以按需选择神经网络模型，并使用自己的私有数据集从大型教师模型中蒸馏知识。为克服资源受限的用户设备训练大型教师模型所面临的挑战，本文采用数字孪生（DT），使教师模型能够在位于服务器且拥有充足计算资源的数字孪生上进行训练。在模型蒸馏过程中，每位用户可以在物理实体或数字代理上更新其模型参数。本文将用户的模型选择、训练卸载和资源分配联合问题建模为混合整数规划（MIP）问题。为求解该问题，本文联合采用Q学习和优化方法：Q学习为用户选择模型并确定是在本地还是服务器上训练，优化方法则根据Q学习的输出为用户分配资源。仿真结果表明，所提出的数字孪生辅助知识蒸馏框架和联合优化方法能够显著提高用户的平均准确率，同时降低总时延。

关键词： 联邦学习；数字孪生；知识蒸馏；异质性；Q学习；凸优化

收稿：xxx**修订：**xxx**编辑：**xxx

一、引言

如今，联邦学习因能够在保护数据隐私的同时，间接利用整个数据集训练出准确率较高的深度学习（DL）模型而备受研究者关注 [1]。由于能够保护用户的数据隐私，联邦学习广泛用于数据敏感领域的神经网络（NN）训练，例如金融数据预测和医疗数据分析。此外，在联邦学习系统中，用户只需将神经网络的参数发送给服务器或特定用户进行聚合，即可提升神经网络的性能；服务器无需收集所有用户的数据，因此，当使用大型数据集训练高性能神经网络时，可减轻服务器存储容量方面的压力。此外，神经网络参数的大小通常远小于训练数据，因此，在通信状态较差的环境中，使用联邦学习仅传输神经网络参数，有助于减少通信时延导致的总训练时延增加 [2]。通过适当的资源管理和训练方法，还可以进一步降低训练成本 [3–5]。

然而，上述所有工作都假设联邦学习中的神经网络模型是同构的，而在现实世界中，设备类型多种多样，只有少数设备是同构的。不同用户可能拥有不同的计算资源、不同的数据，甚至不同的目标，因此需要根据用户的特征提供有针对性的服务 [6, 7]。用户的异质性导致

要对所有用户使用相同的神经网络，会面临诸多挑战[8–11]。首先，计算资源的异质性导致联邦学习系统中的训练时延与准确率之间产生矛盾。在联邦学习中，服务器需要聚合所有用户神经网络的参数，以提升神经网络的性能。因此，如果所有用户都训练大型模型，计算资源有限的用户就会过度拖慢系统的训练时延。另一方面，如果所有用户都训练小型模型，尽管训练速度有所提升，但随着模型大小的减小，联邦学习的准确率会下降。其次，数据异质性可能导致不同用户的数据不再满足独立同分布（非独立同分布）。非独立同分布数据集会带来神经网络收敛方面的挑战，并导致神经网络性能下降，因为在深度学习中，通常假设数据服从独立同分布 [12]。另一个最容易被忽视但十分重要的挑战，是用户目标的异质性。显然，用户参与联邦学习系统的目的是提升自身神经网络的性能。然而，以往大多数工作侧重于提升所有用户数据集上的平均准确率，这可能会导致用户在自己的数据集上的准确率下降。因此，联邦学习系统需要在提升神经网络在所有用户数据集上的平均准确率的同时，提高其在用户自身私有数据集上的准确率。

在实际部署场景中，计算资源的异质性会导致联邦学习出现拖尾效应，从而严重降低训练效率。这是因为不同设备的计算能力差异显著，计算能力强的客户端必须等待计算能力弱的客户端，才能进行模型聚合，导致计算资源大量浪费。因此， [13] 采用分割学习（SL），使计算资源较少的用户只更新神经网络（NN）的少数层，而其他用户则更新神经网络的更多层。显然，分割学习通过在用户之间传输神经网络的中间结果，增加了通信负担。

除了用户计算资源的异质性，数据异质性，尤其是非独立同分布数据集，也备受关注。异构数据会导致不同模型的权重优化方向各异，这是准确率下降的主要原因。因此，基于数据的方案——通过修改数据分布来提升模型性能。Zhao 等人提出使用少量均匀分布的共享数据对全局模型进行预训练，并将共享数据分发给客户端以训练本地模型[14]。实验表明，即使只有极少量共享数据，也能显著提升模型准确率。然而，这种修改数据分布的方法在实际中往往难以实施，因为很难获取均匀分布的共享数据，而且构建共享数据会增加用户隐私泄露的风险。此外，用户可以基于客户端数据微调全局模型，从而获得在本地数据集上表现良好的个性化模型。Fallah 等人提出使用无关元学习来获取初始全局模型，用户只需基于自己的数据进行少量训练即可获得较高准确率 [15]。Arivazhagan 等人提出在客户端模型中使用个性化层，从而大幅提升模型对本地数据集的适应性 [16]。Ma 等人通过辨别不同客户模型中各网络层的重要性，实现了个性化模型聚合 [17]。尽管这些方法在非独立同分布数据上取得了良好效果，但它们都会增加训练开销，且无法实现模型框架的个性化。

为共同解决计算资源、数据和目标方面的异质性，一些研究提出在联邦学习系统中使用知识蒸馏（KD）。作为一种很有前景的技术，知识蒸馏已在许多领域取得成功，因为它可以将大型且训练充分的神经网络（称为教师模型）中的知识迁移到任意小型神经网络（称为学生模型）中：将大型神经网络的输出作为软标签提供给学生模型，并将这些软标签作为额外梯度来训练学生模型。因此，学生模型的性能可以得到提升。由于知识蒸馏方法允许使用任意神经网络作为学生模型，联邦学习系统中的不同用户便可按需选择不同模型。因此，使用知识蒸馏方法可以避免联邦学习训练过程中的拖尾现象。由于不同用户使用的神经网络并不相同，因此无需直接聚合不同用户的神经网络；即使计算资源较多的用户多执行了几个训练轮次，不同用户的神经网络之间也不会出现梯度混淆。此外，在知识蒸馏中，仅需

传输教师模型的输出，因此传输负担远小于分割学习 [18]。Li等人将迁移学习与知识精炼相结合，用于个性化联邦学习，每位客户都可以使用私有数据和特定模型进行训练。Zhou 等人提出了蒸馏单轮联邦学习（DOSFL），从而在确保模型准确率的同时显著降低通信成本 [19, 20]。尽管知识蒸馏允许用户根据需求选择不同模型，但知识蒸馏需要一个大型教师模型，而训练此类模型通常依赖海量数据集。因此，仅使用单个用户有限的数据集无法充分训练教师模型。然而，教师模型通常过于庞大，无法在用户的中央处理器上训练。因此，不能简单地使用联邦学习方法来训练教师模型。因此，先前的研究 [19][21] 主要依赖服务器上的公开数据集，在服务器上训练教师模型。此外，如果教师模型过于庞大，用户运行教师模型来蒸馏知识将面临挑战：一方面，有限的内存资源难以存储教师模型的大量参数；另一方面，在用户的中央处理器上运行教师模型也会产生较大的时延。

幸运的是，不断发展的数字孪生（DT） [22]技术可以解决没有公开数据集以及用户无法运行教师模型这两个难题。在数字孪生中，服务器凭借充足的计算资源，为物理空间中的所有实体及其信息建立名为孪生体的数字副本，以帮助改进物理空间的策略 [23–27]。为了保护物理实体的隐私，未经授权，只有物理实体才能访问其对应孪生体的信息，而不能访问数字空间中其他物理实体的孪生体信息；未经相应物理实体授权，孪生体之间也不能互相访问信息。因此，用户可以安全地将自己的数据集和神经网络参数复制到孪生体中。此外，由于服务器拥有充足的计算资源，教师模型可以存储并运行在服务器上。最初，教师模型可以利用用户孪生体中的数据，通过联邦学习进行训练。随后，用户可以向服务器购买计算资源，获取教师模型的输出，并以知识蒸馏方法训练自己的学生模型。此外，为了应对用户目标的异质性，每个用户只使用自己的私有数据集训练学生模型。因此，学生模型在所有数据集上的平均准确率不仅能通过从教师模型学习通用知识而得到提升，而且，由于蒸馏教师模型知识时仅使用各用户自己的私有数据集，学生模型在其自身数据集上的准确率也会显著提高。

由于用户可能会选择在本地训练小型模型，以降低购买计算资源的成本，因此，为不同用户选择合适的学生模型至关重要，这有助于权衡神经网络的准确率、训练时延和训练成本。随后，应根据所选模型确定是在本地还是服务器上训练学生模型，并购买适当的资源，以降低训练时延和训练成本。合理的调度策略 [28–30]能够有效降低时延和资源消耗。显然，为所有用户选择学生模型并进行卸载属于离散优化问题，而资源管理属于连续优化问题。传统的凸优化方法难以求解这类混合整数问题。尽管Q学习在离散优化问题中取得了成功，并且已被证明，在动作空间有限的离散优化问题中，经过足够数量的训练轮次后，总能达到贝尔曼最优解 [31, 32], 但对于动作空间连续的问题，Q学习通常表现不佳，甚至无法收敛。

因此，本文提出数字孪生辅助知识蒸馏框架，使不同用户能够按需选择不同的学生模型；随后，采用结合Q学习与凸优化的方法，确定模型选择、卸载和资源管理。值得注意的是，为了耦合Q学习与凸优化，在训练过程中，Q学习的奖励由凸优化的输出确定。本文的主要贡献总结如下。

1) 提出了一种面向异构联邦学习的数字孪生辅助知识蒸馏框架。在该框架中，用户可以选择不同的神经网络模型，并在本地训练模型，也可以向服务器购买计算资源，按需训练模型，以应对计算资源的异质性。在用户模型的训练过程中，仅使用每个

用户私有数据集来蒸馏教师的知识，因此用户模型不仅能够学习从教师模型中蒸馏出的通用知识，还能专注于提升其在用户私有数据集上的准确率，从而应对数据和目标的异质性。

2）针对每个用户的模型选择、训练卸载和资源分配联合问题被构建为一个混合整数规划（MIP）问题。为高效求解该问题，本文提出了一种结合Q学习和凸优化的联合优化方法。Q学习用于选择学生模型，并确定是在本地还是服务器上进行训练。基于Q学习的决策，凸优化用于为用户分配资源。值得注意的是，在Q学习的训练过程中，凸优化输出的目标函数值被用作奖励来训练Q学习。

3）我们通过大量仿真评估了所提方法的性能。结果表明，所提联邦学习框架在异构系统中的性能显著优于典型的联邦学习方法。结果还表明，所提出的联合优化方法能够降低联邦学习训练过程中的时延以及传输和计算资源的成本，同时确保每个用户的学生模型性能。

本文其余部分安排如下。在第II 节中，介绍数字孪生辅助的知识蒸馏驱动的联邦学习训练框架，并构建为所有用户选择模型及进行资源管理的问题。第III 节介绍用于解决第II节问题的凸优化辅助Q学习方法，并详细介绍知识蒸馏训练方法。第IV 节评估所提方法的性能，第V节总结全文。

II. 系统模型与问题构建

2.1 系统框架

如图1所示，所提出的异构联邦学习系统数字孪生辅助知识蒸馏框架由两个空间组成：数字空间和物理空间。物理空间包括 N 个用户，每个用户 i配备一台频率为$f_{loc的i}$ 中央处理器（CPU），以及一个 $\mathcal{D}_i$私有数据集；每个用户 i的目标是以较低成本高效训练一个大小为$\theta_{s的i}$学生神经网络模型，并获得较高的准确率。在数字空间中，有一个大小为 θ_t 的教师模型，以及面向用户的 K 孪生体；每个用户孪生体都拥有用户数据集的副本和用户学生模型的参数。数字空间运行在一台CPU频率为 f_{ser} 的服务器上。训练前，首先通过联邦学习方法，利用数字空间中的数据集副本训练教师模型，以保护数据集副本的隐私。随后，在每个训练轮次中，用户模型基于其私有数据集进行训练，从教师模型中蒸馏知识，从而提升学生模型的性能。假设 f_{ser} 远大于任何$f_{loc,\ i}$ 且用户可以从服务器购买计算资源，那么用户可以在本地或服务器上训练学生模型。如果学生模型在本地训练，服务器应将教师模型的输出传送给用户，以便蒸馏教师模型中的知识。否则，学生模型的数字副本将在服务器上训练，此时可以忽略教师模型输出的传输，因为二者都位于同一服务器上。因此，为确保数字空间与物理空间中的模型参数保持一致，每个训练轮次结束后，数字空间或物理空间都应将学生模型的参数传送至物理空间或数字空间，具体取决于学生模型是在本地还是服务器上训练。

由于$f_{loc,i}$ 可能各不相同，简单地使用同一种神经网络会导致准确率与训练时延之间的矛盾。为解决这一矛盾，我们采用知识蒸馏在不同神经网络之间共享知识。由于在知识蒸馏中，教师模型的知识可以蒸馏到深度和宽度任意的学生模型中，因此不同用户可以按需选择不同规模的神经网络。为了突出不同神经网络规模对联邦学习系统性能和计算时延的影响，本文没有像神经网络架构搜索（NAS） [33],那样专门设计神经网络的深度和宽度，而是选取若干具有代表性的神经网络架构组成集合 $\mathcal{M}$，其中 $\mathcal{M}$ 中的每个元素都表示具有特定深度和宽度的神经网络，用户可以在$\mathcal{M}$中按需选择神经网络。

为应对数据和目标的异质性，首先使用联邦学习方法中所有用户在数字空间中的数据副本训练教师模型，

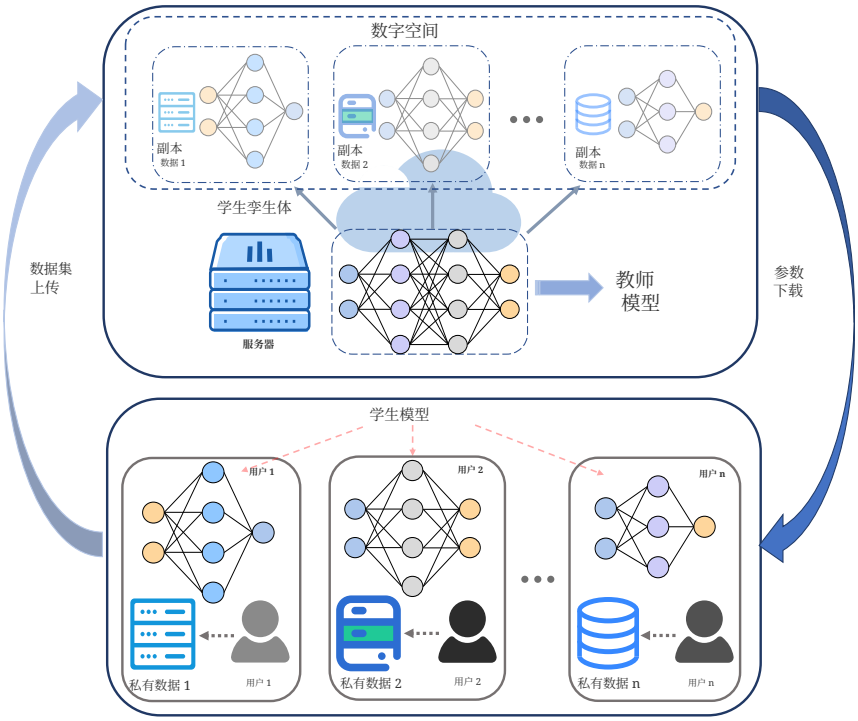


图 1. 联邦学习中的数字孪生辅助知识蒸馏框架。 利用数字孪生在数字空间中训练一个强大的教师模型，随后将教师模型的副本用于对每个学生模型进行蒸馏。在训练过程中，教师模型仅位于数字空间，而学生模型位于用户侧的物理空间，并且数字空间中也有一份学生模型的副本。在物理空间中计算学生模型时，需要定期接收教师模型的特征标签，并同步数字空间中的孪生体。用户私有数据只能在本地或孪生体中访问，不能共享。

这样教师模型便能从所有数据集中提取通用知识。随后，在训练每个用户模型的过程中，仅使用用户私有数据集从教师模型中蒸馏知识。因此，用户模型不仅能从教师模型中学习所有用户数据集的通用知识，还能专注于从用户私有数据集中学习个性化知识，因为训练过程中只使用私有数据。

通常，在知识蒸馏中，庞大的教师模型会预先充分训练。然而，由于用户计算资源的异质性，本文假设教师模型过于庞大，用户无法训练。受数字孪生（DT）特性的启发，本文提出一种数字孪生辅助的知识蒸馏框架，以支持教师模型的训练，并为不同用户选择不同模型。根据 [34], 数字孪生系统可以为物理空间中的所有实体和数据建立一个数字副本，称为孪生体。因此，用户可以在数字空间中创建数据备份。同时，为保护用户数据隐私，数字空间中的孪生体不得与其他孪生体交换用户的私人数据，也不得与物理空间中的其他用户共享数据。不过，为了间接共享数据，允许在孪生体中使用数据训练神经网络，因为在训练过程中，数据始终存储在孪生体中；训练完成后，其他用户和孪生体只能获取具有已训练参数或梯度的神经网络。因此，可以在整个数据集上训练教师模型，以获得较高的性能。随后可以使用知识蒸馏，让教师模型为用户端的小型神经网络（称为学生模型）提供额外的梯度，从而提升学生模型的性能。

在数字空间中，孪生体不得与其他孪生体交换用户的私人数据，也不得与物理空间中的其他用户共享数据。不过，为了间接共享数据，允许在孪生体中使用数据训练神经网络，因为在训练过程中，数据始终存储在孪生体中；训练完成后，其他用户和孪生体只能获取具有已训练参数或梯度的神经网络。因此，可以在整个数据集上训练教师模型，以获得较高的性能。随后可以使用知识蒸馏，让教师模型为用户端的小型神经网络（称为学生模型）提供额外的梯度，从而提升学生模型的性能。

本文为强调针对不同用户进行模型选择和资源管理的重要性，假设数字孪生系统已预先构建，且数字孪生中的孪生体已拥有用户数据的副本，教师模型也已完成训练。根据知识蒸馏系统中的 [35], , 学生模型需要利用教师模型的计算输出或中间结果来更新其参数。

然而，由于教师模型过大，无法存储在用户设备中，因此教师模型始终存储在服务器上，用户需要向服务器购买计算资源，以获取教师模型的输出。假设服务器配备了具有充足计算资源的中央处理器，如 f_{ser} 所示，且 f_{ser} 远高于任何用户中央处理器的频率。为使服务器能够同时为所有用户提供服务，服务器的计算资源可以划分，并且服务器可以同时与所有用户通信。假设服务器通过无线网络与用户通信。因此，受 OFDM 在 4G 和 5G 网络中成功应用的启发，本系统采用 OFDM，以在最大带宽 b_{max} 下避免用户间的干扰。因此，用户 i 与服务器之间的传输速率为

$$r_i = b_i \log_2(1 + \frac{p_i h_i}{n_0}), \tag{1}$$

其中，b_i 是服务器与用户 i之间传输信息所使用的带宽，h_i 是用户 i 与服务器之间的信道增益，p_i 是用户 i的传输功率，n_0 是噪声功率。

为了充分利用用户和服务器的计算能力，用户可以根据需要选择使用本地计算能力或服务器计算能力来训练自己的学生模型。无论用户选择哪种方法来训练学生模型，训练开销都由时延和资源价格构成。如果用户使用自身的计算能力训练学生模型，则在训练过程中，用户需要先等待服务器将教师模型计算出的结果传输给用户。然后，用户使用教师模型的结果和数据集的标签，通过反向传播算法在本地更新学生模型的参数。由于数字孪生系统中的孪生体特征需要与物理空间中实体的特征保持一致，用户还需要将更新后的学生模型参数传输给服务器，以确保数字空间和物理空间中的实体特征保持一致。因此，本地训练学生模型的时延包括教师模型的计算时延 T_{tea}，用户获取教师模型计算结果的传输时延 T_{label}、更新学生模型参数的计算时延 T_{stu}以及在物理空间和数字空间中同步模型参数的传输时延 T_{model}。根据 [36], 对于深度和宽度确定的神经网络，其计算时间与CPU频率成反比。因此，如果用户 i 从服务器购买大小为 f_i 的计算资源，则教师模型参数的计算时延、在用户端更新学生模型的时延 i 且为

$$T_{tea,i} = \frac{\mu_t}{f_i}, \tag{2}$$

$$T_{stu,i} = \frac{\mu_{s,i}}{f_{loc,i}}, \tag{3}$$

其中，μ_t 和 $\mu_{s,\ i}$ 分别是教师模型的计算时间，以及使用单位计算资源更新学生模型参数所需的时间。假设用户 i 处教师模型输出和学生模型参数的大小分别为 θ_l 和 $\theta_{s,\ i}$，传输时延

$$T_{label,i} = \frac{\theta_l}{r_i}, \tag{4}$$

$$T_{model,i} = \frac{\theta_{s,i}}{r_i}. \tag{5}$$

如果用户选择使用服务器算术训练学生模型，由于教师模型和学生模型都位于服务器上，传输教师模型输出的时延不计入总时延。在服务器更新学生模型的参数后，需要将其下载到本地端，因此传输时延仅包括 $T_{model,\ i}$。此外，学生模型的参数更新时间变为$T_{stu,\ i} = {}^{\mu_s}{}_{\cdot\ f_i}{}^{i}$。

2.2 问题构建

为建立问题模型，我们定义两个离散优化变量$\mathbf{x}$和$\mathbf{m}$。$\mathbf{x}$ 表示学生模型是在本地还是服务器上训练，$\mathbf{m}$ 表示用户选择的学生模型。综上，本文的问题可表述如下。

问题1。

$$\min_{\mathbf{m},\mathbf{x},\mathbf{f},\mathbf{b}} \quad \sum_{i=1}^{N} \alpha(T_{tea,i} + T_{stu,i} + T_{model,i} + x_i T_{label,i}) + \beta\left(T_{tea,i} f_i + (1-x_i)\, T_{stu,i} f_i\right) + \delta b_i - \eta_o Acc_{own,i} - \eta_a Acc_{avg,i}, \tag{6}$$

$$s.t. \quad x_i \in \{0,1\}, \tag{7}$$

$$m_i \in \{1,2,\cdots,|\mathcal{M}|\}, \tag{8}$$

$$\sum_{i=1}^{N} b_i \le b_{max}, \tag{9}$$

$$\sum_{i=1}^{N} f_i \le f_{ser}, \tag{10}$$

$$T_{tea,i} = \frac{\mu_t}{f_i}, \tag{11}$$

$$T_{stu,i} = \begin{cases} \mu_{m_i}/f_i & x_i = 0, \\ \mu_{m_i}/f_{loc,i} & x_i = 1. \end{cases} \tag{12}$$

$$r_i = b_i \log_2(1 + \frac{p_i h_i}{n_0}), \tag{13}$$

$$T_{label,i} = \frac{\theta_l}{r_i}, \tag{14}$$

$$T_{model,i} = \frac{\theta_{s,m_i}}{r_i}, \tag{15}$$

$$Acc_{own,i} = \frac{1}{|D_i|}\mathbb{I}(y = NN_{m_i}(in)), \tag{16}$$

$$Acc_{avg,i} = \sum_{i=1}^{N} \frac{1}{|D_i|}\mathbb{I}(y = NN_{m_i}(in)), \tag{17}$$

其中 α β δ和 η_o η_a 为常数因子， μ_{m_i} 表示用户 i 使用单位计算资源更新学生模型参数所需的时间。方程（（6））的第一部分是训练学生模型的总时延；当学生模型在本地训练时， x_i 为 0，因此教师模型输出$T_{label的传输时延i}$ 为 0。第二部分是从服务器购买计算资源的成本；当学生模型在本地训练时，$(1-x_i)T_{stu和i}f_i$ 为 0，这意味着用户只需支付获取教师模型输出所需的计算资源费用和带宽费用。方程（（6））的第三部分是学生模型在其自身数据集及所有用户的整体数据集上的测试准确率；由于我们希望最大化模型准确率，因此最小化准确率的负值。约束（（7））表示学生模型可以在本地或服务器上进行训练。约束 ((8))确保用户只能从模型集中选择学生模型 $\mathcal{M}$。约束 ((9))和 ((10))表明，所有用户购买的计算资源和带宽资源总和不得超过最大计算资源和最大带宽资源。约束 ((11))表示教师模型的计算时延，而约束 ((12)) 中的学生模型训练时延取决于模型是由用户还是服务器训练的。约束 ((13))-((15))表示教师模型输出和学生模型参数的传输时延。约束 ((16))-((17))是评估学生模型准确率的方法，其中 $\mathbb{I}(*)$ 是指示函数，神经网络m_i 是用户 i的学生模型， in 是数据集中某个样本的输入，而 y 是该样本的标签。

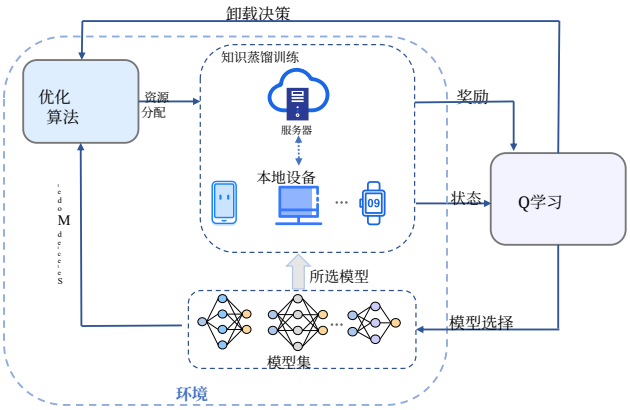


图2。所提方法概述。该方法由三个部分组成：知识蒸馏驱动的模型训练、基于强化学习的模型选择与卸载，以及基于优化的资源分配。*Q*学习负责选择客户模型并做出卸载决策，随后通过凸优化为用户分配资源。使用知识蒸馏完成模型训练后，即可获得奖励。

III. 所提方法

3.1 凸优化辅助Q学习方法

显然，问题1是一个混合整数规划（MIP）问题，因此很难使用传统的凸优化方法高效地优化含有离散变量的问题。由于我们可以同时为所有用户选择学生模型，因此为当前用户选择学生模型

不会影响为其他用户选择学生模型。因此，与[37], 类似，学生模型选择可以建模为一个没有状态转移概率的马尔可夫决策过程（MDP）。尽管Q学习在离散优化问题中无论效率还是性能都取得了成功，但要用它求解含连续优化变量的问题仍然很有挑战性。遗憾的是，在问题1中，$\mathbf{x}$、$\mathbf{m}$ 是离散变量，而$\mathbf{f}$、$\mathbf{b}$ 是连续变量，因此仅使用凸优化或Q学习都难以求解该问题。尽管存在许多基于深度神经网络（DNN）的强化学习方法，例如深度确定性策略梯度（DDPG）[38] 和近端策略优化（PPO）[39], ，能够联合优化离散变量和连续变量，但训练DNN可能耗时较长。没有必要先训练基于DNN的强化学习模型，再用该DNN模型优化联邦学习（FL）的训练。此外，基于DNN的方法推理并获取优化变量所需的时间也很长，在时延敏感型任务中不能简单地忽略 [40], ，从而增加总处理时间。对于Q学习，我们还可以将Q表存储为哈希表。这样，我们可以通过Q学习获得解，其计算复杂度为 $\mathcal{O}(1)$ [41]。因此，优化所造成的时延可以忽略不计。基于此，本文采用基于Q学习的方法来优化该问题。幸运的是，进一步分析问题1可知，在给定$\mathbf{x}$ 和$\mathbf{b}$的情况下，针对$\mathbf{x}$ 和$\mathbf{m}$ 的优化问题是凸问题，而$\mathbf{x}$ 和$\mathbf{m}$都是可通过Q学习进行优化的离散变量。因此，问题1可以分解为两个子问题：1）为每个用户选择学生模型，并确定采用用户算术还是服务器算术进行训练，这可以通过Q学习求解；2）为每个用户分配适当的计算资源和带宽资源，这可以通过凸优化求解。由于Q学习和凸优化都能快速求得解，因此可以忽略算法的推理时间带来的影响。由此，本文采用Q学习求解该问题，具体如下。

问题2。

$$\begin{aligned}\min_{\mathbf{m},\mathbf{x}} \quad & \sum_{i=1}^{N} \alpha(T_{tea,i}+T_{stu,i}+T_{model,i}+x_i T_{label,i}) \\ & +\beta\left(T_{tea,i}f_i+(1-x_i)\,T_{stu,i}f_i\right)+\delta b_i \\ & -\eta_o Acc_{own,i}-\eta_a Acc_{avg,i}, \qquad (18)\\ s.t.\ \ & x_i\in\{0,1\}, \qquad (19)\\ & m_i\in\{1,2,\cdots,|\mathcal{M}|\}, \qquad (20)\\ & \textit{constraint}\ ((11))-((17)), \qquad (21)\end{aligned}$$

Q学习的相关函数定义如下

奖励：我们期望在Q学习完成动作选择后，优化算法能够获得最优目标函数值，因此将方程((18)) 用作奖励函数 r。

状态：每个用户设备的算力不同，导致训练速度各异。此外，客户端与服务器之间的信道状态会影响数据传输速率。为了尽可能获得最大的系统收益，需要根据用户特征，在时延与准确率之间取得平衡。因此，状态为$s=\{\mathbf{f}_{\text{loc}},\mathbf{h}\}$，其中$\mathbf{f}_{\text{loc}}$ 和$\mathbf{h}$ 分别为CPU频率向量和信道状态向量。

动作：由于Q学习执行模型选择和训练调度，动作是 $a=\{\mathbf{x},\mathbf{m}\}$。

据此，Q表更新规则可表示为

$$\begin{aligned}Q(s,a)\leftarrow & Q(s,a)\\ & +\alpha\left[r+\gamma\max_{a'}Q(s',a')-Q(s,a)\right],\end{aligned} \qquad (22)$$

其中，s'、a' 分别表示下一时刻的状态和动作信息，α、γ 分别表示学习率和衰减因子。

然后，基于通过 Q学习优化的变量$\mathbf{x}$ 和$\mathbf{m}$ ，优化$\mathbf{f}$ 和$\mathbf{b}$的子问题是一个凸优化问题，如下所示。

问题3。

$$\min_{\mathbf{m},\mathbf{x}} \quad \sum_{i=1}^{N} \alpha(T_{tea,i} + T_{stu,i} + T_{model,i} + x_i T_{label,i}) + \beta\left(T_{tea,i} f_i + (1 - x_i) T_{stu,i} f_i\right) + \delta b_i, \tag{23}$$

$$s.t. \qquad constraints\ ((9)) - ((15)). \tag{24}$$

与问题1和问题2不同，目标函数（（23））不包含神经网络性能。这是因为每个用户的神经网络性能仅与所选学生模型的结构有关，而计算资源和带宽资源的优化只会影响训练速度，不会影响神经网络性能。显然，问题3是凸的，因此我们可以通过凸优化来求解。

因此，如图2 所示，Q学习首先为所有用户选择学生模型，并确定学生模型是在本地还是服务器上训练。然后，根据所选学生模型和卸载决策，采用凸优化为所有用户分配资源。随后，可以计算目标函数（（6））的值，并将其用作训练Q学习的奖励。

3.2 数字孪生辅助的教师与学生模型训练方法

知识蒸馏（KD）是一种基于教师-学生网络结构的神经网络模型压缩技术。由于其出色的性能以及与其他深度学习技术的高度兼容性，近年来得到了广泛应用。通常，不同规模模型的性能存在差异。参数更多且计算复杂度更高的模型，能够比小型模型获得更高的准确率。然而，受设备计算能力和存储容量的限制，以及特定任务需求的影响，许多情况下无法部署大型模型。因此，知识蒸馏可在尽可能降低准确率损失的同时压缩模型，也可视为利用大型模型提升小型模型性能的一种方式。

应用知识蒸馏需要一个强大的教师模型。训练教师模型需要强大的计算能力和内存，客户端无法满足这些需求。因此，我们使用数字孪生技术将用户数据复制到数字域，并在服务器上训练教师模型。我们使用FedSGD [42]训练教师模型，并在每个用户完成一批数据的计算后聚合梯度，以更新全局模型。由于用户的私有数据具有高度异质性，我们将批次大小设为私有数据集的大小，以增强模型收敛的稳定性和准确率。这样，每次参数更新都会使用全部用户数据，从而避免因聚合不同优化方向上的模型参数而导致性能下降。需要注意的是，尽管这种方法能够在保障数据安全的同时获得准确率很高的教师模型，但它会消耗大量计算资源和存储空间。

传统知识蒸馏方法使用教师模型的输出作为软目标，以帮助优化学生模型的参数。具体而言，学生网络的损失函数可表示为

$$L_{\mathrm{KD}} = L_{\mathrm{hard}} + T^2 L_{\mathrm{soft}}, \tag{25}$$

其中，T 为温度系数。L_{hard} 表示传统交叉熵损失，L_{soft} 是根据教师模型和学生模型的预测对得到的软目标

$$\begin{aligned} L_{\mathrm{hard}} &= L_{\mathrm{CE}}\left(\boldsymbol{p}_s, \boldsymbol{y}\right) \\ L_{\mathrm{soft}} &= L_{\mathrm{KL}}\left(\boldsymbol{p}_s, \boldsymbol{p}_t\right), \end{aligned} \tag{26}$$

其中，$L_{\mathrm{CE}}(\cdot)$ 为交叉熵函数，$L_{\mathrm{KL}}(\cdot)$ 表示库尔贝克-莱布勒散度，$\boldsymbol{y}$ 为标签向量，$\boldsymbol{p}_s$和 $\boldsymbol{p}_t$ 分别为学生模型和教师模型的预测向量。与仅使用硬损失函数的传统单模型训练方法相比，教师模型提供的软目标能为学生模型提供更多知识，从而提高学生模型的准确率。

尽管知识蒸馏在许多领域取得了巨大成功，但教师模型与学生模型的性能之间仍存在显著差距，而且学生模型的性能提升很大程度上依赖于合适的模型框架和合理的超参数值。为进一步提升学生模型的性能，已有多种

先进的蒸馏方案。这些方案能够进行更深入的知识提取，并采用更精细的知识表示方法[43–45]。然而，这些方案对模型框架有严格要求，训练过程也往往更加复杂。使用同一个教师模型蒸馏多个差异显著的学生模型，难以取得令人满意的性能；而针对不同学生模型训练多个教师模型，则会带来显著的额外开销。此外，对于训练难度较高的复杂任务，仅依靠软目标进行蒸馏所获得的收益并不充分。因此，我们采用一种更强大而简单的蒸馏方法（simKD），复用教师模型的线性输出层，以进行更深入的知识提取[46]。

通常，分类模型依赖于由多个层次结构组成的编码器，这些结构经过精心设计并相互连接，以进行特征提取，从而将图像的低级特征转换为高级特征，最后通过线性层输出分类结果。模型需要具备这种特征提取能力，才能处理各种图像；因此，在特定数据集上训练的模型只需微调参数，便可迁移到其他数据集并取得良好性能。模型的分类准确率在很大程度上取决于其特征提取能力，因此现有的大多数工作都集中于设计更高效的特征提取结构。对于不同的模型，其大部分计算量都来自特征提取部分，而作为分类器的线性层所占比例很小。因此，为了尽可能使学生模型的性能接近教师模型，我们在特征方面对学生模型和教师模型进行匹配，并将学生模型的目标函数定义为

$$L_{\text{simKD}} = \|\boldsymbol{f}_t - P(\boldsymbol{f}_s)\|^2, \tag{27}$$

其中，$\boldsymbol{f}_t$、$\boldsymbol{f}_s$分别表示教师模型和学生模型的特征向量，$P(\cdot)$是用于匹配特征向量维度的投影器。为进一步提高学生模型的准确率并降低训练开销，学生模型直接复用教师模型的线性层，以确保其能够根据所学到的特征知识正确执行基于学习到的特征知识进行分类。
由于所选模型的线性层具有相同或相似的结构，学生模型的计算复杂度几乎保持不变。

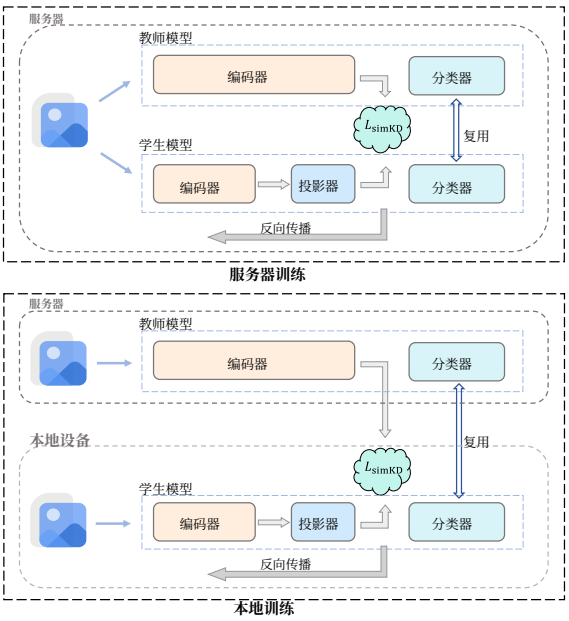


图3. 训练管理。 根据学生模型的计算位置，模型训练有两种场景：在服务器上训练和本地训练。

我们将客户端模型作为学生模型，并在图3中分别展示了在服务器端和本地使用 simKD 训练客户端模型的过程。在服务器上训练时，教师模型和客户端模型使用相同的数据，教师模型的特征作为标签对，用于客户端模型的反向传播。本地训练时，教师模型的前向传播在服务器上完成，然后将特征标签发送给用户，客户端模型的前向和反向传播则在本地完成。

IV. 仿真结果

在本节中，我们通过仿真评估所提出的数字孪生辅助知识蒸馏框架，以及基于Q学习凸优化方法的联合资源管理方法的性能。所有用户的传输功率 p 均为0.1 W，最大带宽 b_{max} 为10 MHz。所有用户均随机分布在距离服务器 d 为10至100米的范围内，信道增益 $h = \frac{g_0}{d^\gamma}$ 为

$g_0 = -40$ dB，且 $\gamma = 2.8$。中央处理器频率为10 GHz，用户的中央处理器频率则在 $[0.5, 2]$ GHz内随机取值。仿真中的用户数量 K 为4。为展示所提出的数字孪生辅助知识蒸馏训练方法（知识蒸馏）适应异构数据和设备的能力，我们选择了四种不同规模的模型，以匹配客户端设备。具体而言，采用VGG-8、ResNet-8x4、ResNet-14x4和ResNet-26x4作为学生模型，并采用ResNet-32x4作为教师模型；同时在具有不同分布的数据集上测试其性能。在CPU频率为1 GHz时，学生模型训练一个训练轮次分别耗时6.83秒、8.75秒、12.27秒和18.96秒。

数据集与基线方法。我们采用两种方式对CIFAR-100数据集进行划分：将所有类别的图像分成四个相等的部分，构成四个独立同分布子数据集。按照图像类别划分数据集，每个子数据集包含 25 个类别的图像，从而得到四个相互正交的非独立同分布子数据集。我们将所提出的算法与传统联邦学习[47]（FL）进行比较，同时也与在每个客户端分别训练学生模型的方法（STU）进行比较。值得注意的是，在这三种算法中，知识蒸馏驱动方法和仅训练学生模型的方法能够为不同设备部署合适的模型，而传统联邦学习由于需要进行参数聚合，只能使用一个模型。

训练细节。我们使用随机梯度下降优化器进行训练，并将训练轮次设为 100。对于知识蒸馏，我们将 Nesterov 动量和初始学习率分别设为 0.9 和 0.05，并在总计 100 个训练轮次中的第 45、60、75 和 90 轮将学习率除以 5。小批量大小设为 64，权重衰减设为7.5×10^{-4}。对于联邦学习，我们根据[47]设置参数。STU 使用与知识蒸馏相同的参数。

4.1 数字孪生辅助知识蒸馏训练方法的性能

为了展示不同方法之间的性能差异，我们比较了在不同数据集上训练 VGG-8 所获得的准确率。图 4 和图 5 展示了各算法在独立同分布数据集上训练得到的准确率曲线。可以观察到，知识蒸馏和传统联邦学习都取得了良好的性能，而单独训练的学生模型准确率较低。这是因为前两种方法能够借助强大的教师模型或其他学生模型来更新自身参数，并拓展模型的知识来源，从而有效提升模型性能。相比之下，仅依据自身数据训练学生模型，由于训练数据量不足，会严重限制模型准确率。具体而言，知识蒸馏和联邦学习分别取得了 62.71% 和 59.90% 的准确率，显著高于学生模型 43.23% 的准确率上限；从图中也可以看出，它们的收敛稳定性更好。尽管传统联邦学习在独立同分布数据集上也能取得良好效果，但知识蒸馏驱动的训练方法仍将准确率提高了 2.81%。

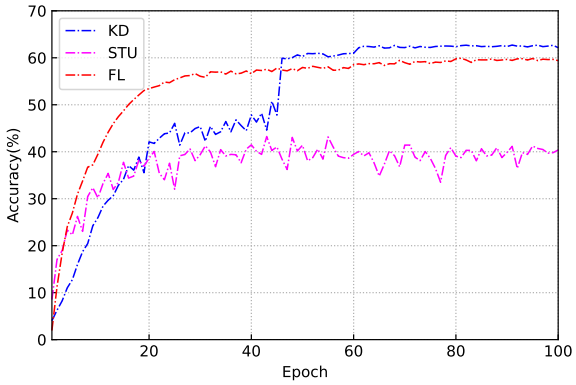


图4. 独立同分布训练集上的测试准确率。

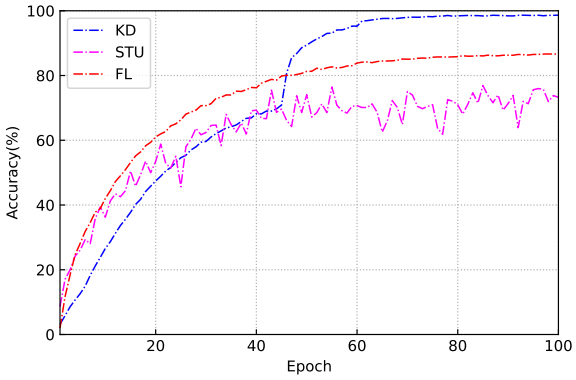


图5. 独立同分布训练集上的训练准确率。

图6 — 图8 展示了非独立同分布数据集上的训练结果。为说明异质数据集对准确率的不利影响，我们分别考察模型在原始测试数据和异质测试数据上的准确率。原始数据集包含全部 100 类测试图像，而异质数据集仅提取与训练数据匹配的 25 类图像。由于传统联邦学习

需要聚合来自不同客户端的模型参数，因此模型在异质测试集上的准确率与其在原始测试集上的准确率相当。显然，联邦学习在异质数据集上完全无法收敛。即使训练 100 个轮次后，分类准确率仍在1%左右，基本与随机分类相同。这一结果与 Sattler 等人的发现一致。[48]尽管联邦学习方法是为联邦学习场景设计的，但它只考虑客户端之间的数据独立同分布。在这种情况下，通常认为各客户端经验风险函数的期望是总体数据期望的无偏估计。上述结果表明，使用联邦学习方法可以取得良好的性能。然而，在非独立同分布情况下，上述假设中的无偏性不再成立。因此，在训练过程中，不同客户端的模型权重会朝不同方向更新。由于传统联邦学习采用权重平均来聚合各模型的参数，非独立同分布的数据分布使其不再收敛 [14]。

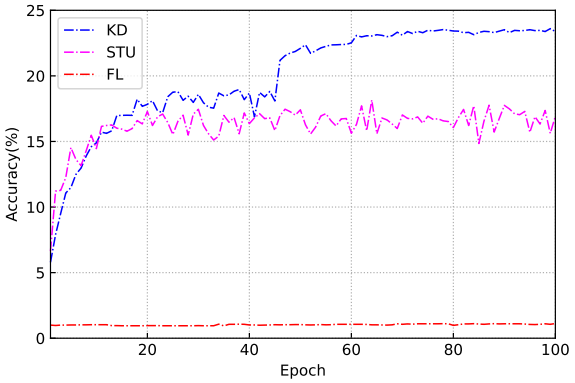


图6. 使用非独立同分布训练集时的完整测试集准确率。

尽管所提出的知识蒸馏驱动方法在非独立同分布数据集上表现出良好的适应性，但与传统知识蒸馏方案不同，simKD旨在使学生模型学习教师模型的特征提取能力。在非独立同分布的情况下，学生模型只有25个训练标签，这意味着其余75张图像的标签对于分类器而言始终为零。这会促使模型在训练期间尽可能降低这75张图像的分类概率，以获得更高的训练准确率。因此，这会导致

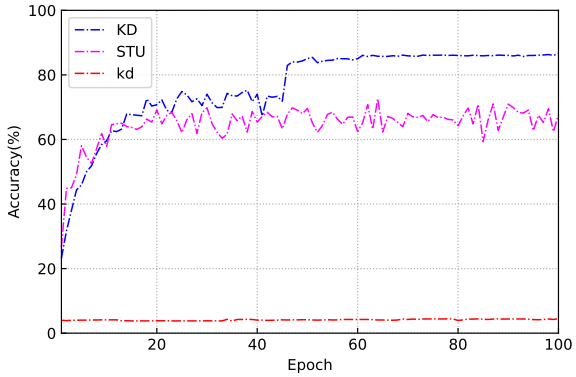


图7. 使用非独立同分布训练集时的私有测试集准确率。

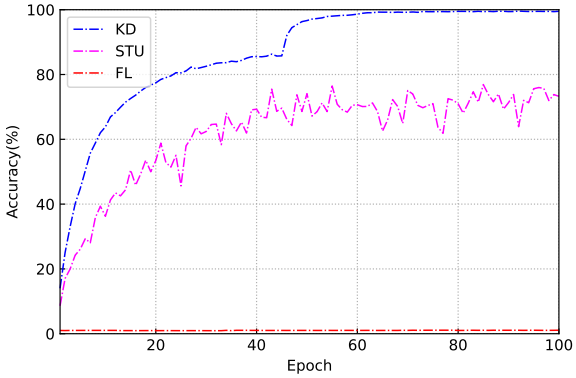


图8. 使用非独立同分布训练集时的训练准确率。

模型在完整测试数据上的准确率显著下降。让学生模型复用教师模型的分类器并进行深度特征对齐，可以在很大程度上缓解这一问题。这是因为在图像识别任务中，从不同图像中提取特征所执行的操作通常相似，而分类器必须针对特定图像类别进行学习。训练集上的准确率表明，知识蒸馏驱动方法能够快速收敛且稳定性良好。可以看出，异构测试集上的准确率显著高于原始数据集上的准确率，这是因为划分数据集后，测试数据与训练数据的分布保持一致，而类别数和图像总数则大幅减少。因此，模型的训练难度降低。在这种情况下，与学生模型相比，知识蒸馏的性能提升了 14%。知识蒸馏和学生模型在原始测试集上的准确率均显著下降。显然，缺少 75 个类别的训练图像严重影响了

表1. 完整数据集上的测试准确率（%）。

方法	KD				FL				学生模型			
训练集	IID		非独立同分布		IID		非独立同分布		IID		非独立同分布	
模型	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
VGG-8	62.71	86.61	23.60	63.51	59.90	84.49	1.12	5.01	43.23	73.83	18.16	23.62
ResNet-8x4	68.13	90.85	25.81	67.88	62.13	87.00	1.47	5.40	45.26	76.53	18.50	24.00
ResNet-14x4	71.84	92.40	28.48	73.43	65.64	88.50	1.43	5.11	49.97	79.29	19.35	24.08
ResNet-26x4	72.55	92.59	28.73	73.51	66.10	89.22	1.23	5.04	49.32	78.66	19.53	24.20

表2. 私有数据集上的测试准确率（%）。

方法	KD		FL		学生模型	
模型	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
VGG-8	86.53	96.75	4.47	20.04	72.59	94.47
ResNet-8x4	89.46	98.11	5.88	21.60	73.95	95.99
ResNet-14x4	90.98	98.11	5.72	20.44	77.36	96.31
ResNet-26x4	90.14	97.75	4.92	20.16	78.08	96.79

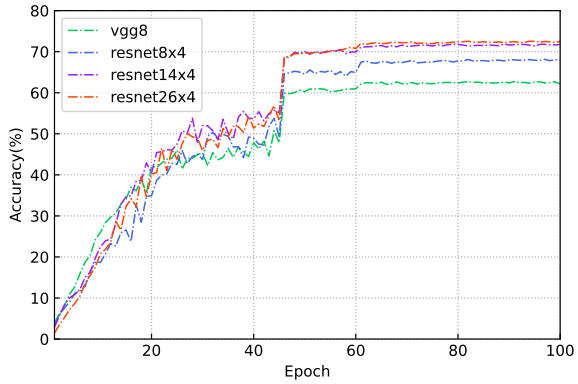


图9. 采用知识蒸馏在独立同分布训练集上训练的不同模型的测试准确率。

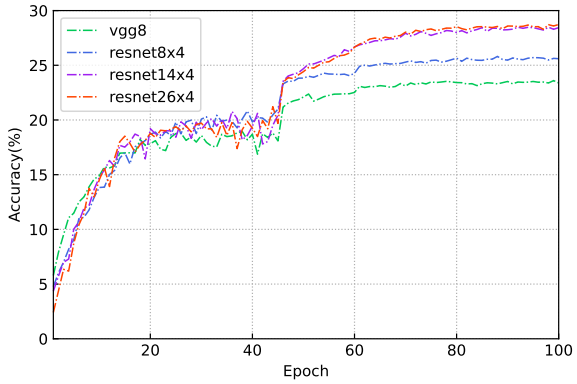


图10. 采用知识蒸馏在非独立同分布训练集上训练的不同模型的测试准确率。

模型的分类能力，而图 6 与图 7中的结果差异巨大，这表明模型难以识别训练集中缺失类别的图像。在这种情况下，学习教师模型的知识可以有效弥补训练数据缺失带来的不利影响，与学生模型相比，性能提升了 5.4% 。

知识蒸馏不仅提升了客户端模型的性能，还能很好地适配不同规模的学生模型。图9 和图10 展示了采用知识蒸馏训练的不同客户端模型的测试准确率。可以观察到，每个模型都能收敛得很快，并稳定在较高的准确率。不同模型最终达到的准确率有所不同，这取决于模型本身的表征能力。此外，与使用非独立同分布数据训练相比，使用独立同分布数据训练的模型收敛得更快。这是因为两种数据中的图像数量相同，而异质训练集与测试集之间的分布差异增加了训练难度。

表 1 和表2 展示了使用所有方法训练不同客户端模型时的最优准确率。显然，知识蒸馏在所有情况下都取得了最佳性能。相比之下，联邦学习在

独立同分布数据上能够取得良好性能，但无法处理非独立同分布数据。虽然学生模型没有像联邦学习那样遭遇准确率灾难性下降，但其在非独立同分布数据上的表现也不尽如人意。

4.2 Q学习与凸优化联合优化的性能

为了评估所提出的Q学习与凸优化联合优化方法的性能，我们将其与另外三种方法进行比较。仅使用Q学习的方法利用Q学习为所有用户选择学生模型并分配资源；通过与该方法比较，可以验证联合使用凸优化的有效性。FL-Min和FL-Max则为所有用户分别选择相同的最小/最大学生模型，然后使用Q学习确定卸载方法，并使用凸优化分配资源；通过比较这两种方法，可以说明为不同用户选择不同模型的重要性。在表格3 和表格4中，平均时延是系统中 K 个用户训练一个轮次所需时间的平均值，Acc-Own是 K 个学生模型在各自私有数据集上的准确率，而Acc-Avg是 K 个学生模型在所有用户数据集上的平均准确率。我们还统计了不同算法下各学生模型被选中的概率，如表格3和表格4最后四行所示。如表格3 和表格4所示，所提出的联合优化方法始终能取得最佳目标值，因为Q学习允许用户选择最佳学生模型和卸载方法，而凸优化则能在Q学习的基础上找到最佳资源分配方案。此外，根据 [49], 对于具有有限离散动作空间的问题，经过充分的训练轮次后，Q学习总能收敛到贝尔曼最优解。由于 m 和 x 均为离散值，因此从理论上讲，Q学习能够达到贝尔曼最优解。然而，如果使用Q学习联合选择学生模型并分配资源，由于 f 和 b 是连续值，问题的动作空间将变为无限。因此，Q学习很难取得高性能，仅使用Q学习的方法的性能始终低于所提出的联合优化方法。值得注意的是，尽管仅使用Q学习的方法不如联合优化方法，但由于资源管理的作用，在采用最佳资源分配方案时，它始终优于FL-Max和FL-Min。这一现象揭示了为计算资源异构的设备选择不同学生模型的重要性。由于最小模型大小带来的传输时延和训练时延较低，FL-Min在训练一个轮次时的平均时延始终最低。

根据表 1 和表 2，尽管较大的学生模型在所有用户的数据集上平均表现更好，但最大的模型 ResNet-26x4 在用户私有数据上的表现并不优于 ResNet-14x4。这是因为大型模型提取数据集通用特征的能力更强，因此受平均特征的影响，其在特定数据集上的表现可能不如小型模型。由于用户的目标存在异质性，学生模型需要在用户自己的私有数据集上取得良好表现，因此，基于 Q 学习的模型选择算法很少选择最大的 ResNet-26x4 作为学生模型，因为它在用户私有数据集上的准确率较低。

V. 结论

本文研究了联邦学习系统中用户在数据、计算资源和目标方面的异质性问题。我们提出了一种数字孪生辅助的知识蒸馏框架，使联邦学习系统中的不同用户能够选择不同的神经网络模型。随后，我们联合使用 Q 学习和凸优化，为每个用户选择神经网络模型并分配资源。仿真结果验证了所提框架和联合优化方法的有效性与效率。我们的工作有助于深入认识联邦学习系统中异质性这一重要但尚未得到充分探索的领域，并为数字孪生技术提供了有前景的应用方向。未来，我们将重点研究分布式优化方法，以解决模型选择和资源管理问题。

致谢

本研究得到了中国国家重点研发计划

表3. 不同算法在非独立同分布数据集上的性能。

指标	所提方法	仅使用Q学习	FL-Min	FL-Max
目标值	-1.03	-0.99	-0.95	0.0075
平均时延	14.33 秒	15.66 秒	14.02 秒	27.01 秒
Acc-Own	90.19 %	89.72 %	1.12 %	1.23 %
平均准确率	27.10 %	26.52 %	1.12 %	1.23 %
VGG-8 的概率	1.5 %	6.5 %	100 %	0 %
ResNet-8x4 的概率	50.5 %	62.0 %	0 %	0 %
ResNet-14x4 的概率	48.0 %	28.0 %	0 %	0 %
ResNet-26x4 的概率	0.0 %	3.5 %	0.0 %	100 %

表4. 不同算法在独立同分布数据集上的性能。

指标	所提方法	仅使用Q学习	FL-Min	FL-Max
目标值	-1.27	-1.22	-1.11	0.0074
平均时延	13.91 秒	18.09 秒	13.07 秒	20.55 秒
Acc-Own	71.17 %	70.47 %	59.9 %	65.64 %
平均准确率	71.17 %	70.47 %	59.9 %	65.64 %
VGG-8 的概率	0.5 %	0.0 %	100 %	0 %
ResNet-8x4 的概率	17.0	39 %	0 %	0 %
ResNet-14x4 的概率	80.0	49.5 %	0 %	0 %
ResNet-26x4 的概率	2.5 %	11.5 %	0.0 %	100 %

（2020YFB1807700）、中国国家自然科学基金（NSFC）项目（编号：62071356）以及重庆市移动通信技术重点实验室项目（编号：cqupt-mct- 202202）的资助。

参考文献

[1] ZHANG C, XIE Y, BAI H, 等. 联邦学习综述[J]. 《知识型系统》, 2021, 216: 106775.[2] YANG Z, CHEN M, WONG K K, 等. 面向6G的联邦学习：应用、挑战与机遇[J]. 《工程》, 2021.[3] SHEN J, CHENG N, YIN Z, 等. 联邦学习的联合资源分配与用户调度方案[C]//2021 IEEE 第94届车辆技术大会（VTC2021-Fall）. IEEE, 2021: 1-5.[4] BEITOLLAHI M, LU N. 面向节能型6G无线网络中联邦学习的多帧调度[C/OL]//IEEE INFOCOM 2022——IEEE计算机通信

通信研讨会（INFOCOM WKSHPS）。2022：1-6。DOI：10.1109/INFOCOMWKSHPS54753.2022.9798090。[5] WU Y、SONG Y、WANG T 等。无线能量传输辅助的非正交多址联邦学习：一种成本高效的方法[J/OL]。《IEEE通信汇刊》，2022，70(4)：2853-2869。DOI：10.1109/TCOMM.2022.3153068。[6] HUI Y、SU Z、LUAN T H。无人时代：智慧城市中的服务响应框架[J/OL]。《IEEE智能交通系统汇刊》，2022，23(6)：5791-5805。DOI：10.1109/TITS.2021.3058385。[7] HUI Y、CHENG N、SU Z 等。6G异构车联网中的安全个性化边缘计算服务[J/OL]。《IEEE物联网学报》，2022，9(8)：5920-5931。DOI：10.1109/JIOT.2021.3065970。[8] WANG S、LEE M、HOSSEINALIPOUR S 等。异构联邦学习的设备采样：理论、算法与实现

现[C]//IEEE INFOCOM 2021-IEEE计算机通信会议。IEEE，2021：1-10。[9]DENG Y、LYU F、REN J 等。通过质量感知的用户激励和自动加权模型聚合改进联邦学习[J/OL]。IEEE并行与分布式系统汇刊，2022，33(12)：4515-4529。DOI：10.1109/TPDS.2022.3195207。[10] DENG Y、LYU F、REN J 等。Fair：通过精准的用户激励和模型聚合实现质量感知联邦学习[C/OL]//IEEE INFOCOM 2021 - IEEE计算机通信会议。2021：1-10。DOI：10.1109/INFOCOM42981.2021.9488743。[11]DENG Y、LYU F、REN J 等。Auction：面向高效联邦学习的自动化质量感知客户端选择框架[J/OL]。IEEE并行与分布式系统汇刊，2022，33(8)：1996-2009。DOI：10.1109/TPDS.2021.3134647。[12] LI X、HUANG K、YANG W 等。FedAvg在非独立同分布数据上的收敛性[J]。arXiv预印本 arXiv:1907.02189，2019年。[13] THAPA C、ARACHCHIGE P C M、CAMTEPE S 等。SplitFed：当联邦学习遇上分割学习[C]//《人工智能促进会会议论文集》第36卷。2022：8485-8493。[14]ZHAO Y、LI M、LAI L 等。基于非独立同分布数据的联邦学习[J]。arXiv预印本 arXiv:1806.00582，2018 年。[15] FALLAH A 、MOKHTARI A、OZDAGLAR A。具有理论保证的个性化联邦学习：一种模型无关元学习方法[J]。《神经信息处理系统进展》，2020，33：3557-3568。[16]ARIVAZHAGAN M G、AGGARWAL V、SINGH A K 等。具有个性化层的联邦学习[J]。arXiv预印本 arXiv:1912.00818，2019年。[17] MA X、ZHANG J、GUO S 等。面向个性化联邦学习的分层模型聚合[C]//IEEE/CVF计算机视觉与模式识别会议论文集。2022：10092-10101。[18] SINGH A、VEPAKOMMA P、GUPTA O 等。通信效率的详细比较

分割学习和联邦学习的效率[J]。arXiv 预印本 arXiv:1909.09145，2019年。[19] LI D, WANG J. Fedmd：通过模型蒸馏实现异构联邦学习[J]。arXiv 预印本 arXiv:1910.03581，2019年。[20]ZHOU Y, PU G, MA X, 等. 蒸馏单轮联邦学习[J]。arXiv 预印本 arXiv:2009.07999，2020年。[21] MOTHUKURI V, PARIZI R M, POURIYEH S, 等. 联邦学习的安全与隐私综述[J]。Future Generation Computer Systems，2021年，115：619-640。[22] YU Q, REN J, FU Y, 等. 网络孪生：下一代网络架构的起源[J/OL]。IEEE Wireless Communications，2019年，26(6)：111-117。DOI：10.1109/MWC.001.1900184。[23]YIN Z, CHENG N, LUAN T H, 等. 网络孪生赋能的星地融合车联网物理层安全[J/OL]。IEEE Transactions on Vehicular Tech- nology，2022年，71(5)：4561-4572。DOI：10.1109/TVT.2021.3133574。[24]WANG X, MA L, LI H, 等. 数字孪生辅助的边缘任务调度高效强化学习[C/OL]//2022 IEEE第95届车辆技术会议：(VTC2022- Spring)。2022年：1-5。DOI：10.1109/VTC2022-Spring54318.2022.9860495。[25] ZHENG J, LUAN T H, HUI Y, 等. 数字孪生赋能的车联网异构网络选择与知识迁移[J/OL]。IEEE Transactions on Vehicular Tech- nology，2022年：1-16。DOI：10.1109/TVT.2022.3190941。[26] HUI Y, MA X, SU Z, 等. 协作即服务：数字孪生赋能的协作式分布式自动驾驶[J/OL]。IEEE Internet of Things Journal，2022年：1-1。DOI：10.1109/JIOT.2022.3161677。[27] FAN B, WU Y, HE Z, 等. 数字孪生赋能的智能车辆换道移动边缘计算[J/OL]。IEEE Network，2021年，35(6)：194-201。DOI：10.1109/MNET.201.2000768。[28]SUN R, WANG Y, LYU L, 等. 面向时延的D2D移动网络缓存策略[J/OL]。IEEE Transactions on Vehicu- lar Technology，2020年，69(8)：8529-8541。DOI：

10.1109/TVT.2020.2996238。[29]CHENG N, LYU F, QUAN W, 等. 面向物联网应用的空中辅助计算卸载：一种基于学习的方法[J/OL]。IEEE通信领域精选期刊，2019年，37(5)：1117-1129。DOI：10.1109/JSAC.2019.2906789。[30] CHENG N, HE J, YIN Z, 等. 6G服务导向的空天地一体化网络：综述[J]。《航空学报》，2021年。[31]ZHOU Y, MA X, HU S, 等. 基于混合深度强化学习的服务质量体验驱动多无人机网络自适应部署策略[J/OL]。IEEE Internet of Things Journal，2022年，9(8)：5868-5881。DOI：10.1109/JIOT.2021.3066368。[32] WANG X, GUO H. 基于深度强化学习的群体机器人移动性感知计算卸载[C/OL]//2021 IEEE第18届年度消费者通信与网络会议（CCNC）。2021年：1-4。DOI：10.1109/CCNC49032.2021.9369456。[33] ELSKEN T, METZEN J H, HUTTER F. 神经架构搜索：综述[J]。《机器学习研究杂志》，2019年，20(1)：1997-2017。[34] LIU M, FANG S, DONG H, 等. 数字孪生概念、技术与工业应用综述[J]。《制造系统杂志》，2021年，58：346-361。[35] GOU J, YU B, MAYBANK S J, 等. 知识蒸馏：综述[J]。《国际计算机视觉杂志》，2021年，129(6)：1789-1819。[36] LECUN Y, BENGIO Y, HINTON G. 深度学习[J]。《自然》，2015年，521(7553)：436-444。[37] MENG F, CHEN P, WU L. 基于深度Q学习的多用户蜂窝网络功率分配[C]//ICC 2019-2019 IEEE国际通信会议（ICC）。IEEE，2019年：1-6。[38]SILVER D, LEVER G, HEESS N, 等. 确定性策略梯度算法[C]//国际机器学习会议。PMLR，2014年：387-395。[39]SCHULMAN J, WOLSKI F, DHARIWAL P, 等. 近端策略优化算法[J]。arXiv 预印本 arXiv:1707.06347，2017年。[40] MA L, CHENG N, WANG X, 等. 按需

基于知识辅助动态神经网络的6G无线网络资源管理[C/OL]//ICC 2022 - IEEE国际通信会议。2022：1-6。DOI：10.1109/ICC45855.2022.9882279。[41] MAURER W D，LEWIS T G。哈希表方法[J]。ACM计算调查（CSUR），1975，7(1)：5-19。[42]MCMAHAN H B，MOORE E，RAMAGE D，等。使用模型平均进行深度网络的联邦学习[J]。arXiv预印本 arXiv:1602.05629，2016，2。[43] WANG L，YOON K J。视觉智能的知识蒸馏与学生-教师学习：综述与新展望[J]。IEEE模式分析与机器智能汇刊，2021。[44] CHEN D，MEI J P，ZHANG Y，等。采用语义校准的跨层蒸馏[C]//人工智能促进会会议论文集：第35卷。2021：7028-7036。[45]KOMODAKIS N，ZAGORUYKO S。更加关注注意力：通过注意力迁移提升卷积神经网络的性能[C]//ICLR。2017。[46] CHEN D，MEI J P，ZHANG H，等。利用复用教师分类器进行知识蒸馏[C]//IEEE/CVF计算机视觉与模式识别会议论文集。2022：11933-11942。[47] MCMAHAN B，MOORE E，RAMAGE D，等。基于去中心化数据的深度网络通信高效学习[C]//人工智能与统计学。PMLR，2017：1273-1282。[48] SATTLER F，WIEDEMANN S，M¨uLLER K R，等。基于非独立同分布数据的稳健且通信高效的联邦学习[J/OL]。IEEE神经网络与学习系统汇刊，2020，31(9)：3400-3413。DOI：10.1109/TNNLS.2019.2944481。[49] WATKINS C J，DAYAN P。Q学习[J]。《机器学习》，1992，8(3)：279-292。

人物简介



XiuchengWang 于2021年获得西安电子科技大学电信工程学士学位，目前正在西安电子科技大学攻读硕士学位。他的研究兴趣包括数字孪生和无线网络图神经网络。



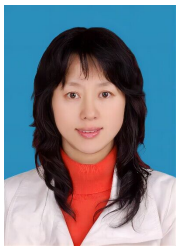
NanCheng Nan Cheng 于2016年获得加拿大安大略省滑铁卢市滑铁卢大学电气与计算机工程系博士学位，并于2009年和分别为 2012 年。他于2017年至2019年在加拿大安大略省多伦多市多伦多大学电气与计算机工程系担任博士后研究员。目前，他是中国陕西省西安市西安电子科技大学综合业务网国家重点实验室和电信工程学院的教授。他目前的研究方向包括B5G/6G、空天地一体化网络、车联网中的大数据以及自动驾驶系统。他的研究兴趣还包括性能分析、媒体访问控制、机会通信以及人工智能在车联网中的应用。



LongfeiMa 于 2022 年获得西安电子科技大学电信工程学士学位，目前正在该校攻读硕士学位。他的研究兴趣包括数字孪生以及无线网络中的动态神经网络。



RuijinSun 于 2019 年获得中国北京邮电大学博士学位。她现任西安电子科技大学电信工程学院讲师，并在综合业务网国家重点实验室工作。她曾作为联合博士后研究员，与 Peng 2019年至2021年在Cheng Laboratory和清华大学。她曾于2017年9月至2018年9月在加拿大滑铁卢大学担任访问学生。她的研究兴趣包括知识驱动的无线资源分配和多输入多输出信号处理。



RongChai 于1995年和1998年分别获得中国成都电子科技大学的工学学士和硕士学位，并于2008年获得加拿大安大略省汉密尔顿麦克马斯特大学电气工程博士学位。2008年，她加入了通信学院，通信与信息工程学院任职，重庆邮电大学（中国重庆），目前她是该校教授。她的研究兴趣包括无线通信和网络理论。



NingLu 于 2007 年和 2010 年分别获得中国上海同济大学电气工程学士和硕士学位，并于 2015 年获得加拿大安大略省滑铁卢滑铁卢大学电气工程博士学位。他目前是该校一名助理教授，任职于.....电气与计算机工程系，女王大学，加拿大安大略省金斯顿。在加入女王大学之前，他曾任加拿大不列颠哥伦比亚省坎卢普斯汤普森河大学计算机科学系助理教授。2015年至2016年，他在美国伊利诺伊州尚佩恩的伊利诺伊大学协同科学实验室担任博士后研究员。2009年夏天，他还曾在日本东京的国立情报学研究所实习。他在多份顶尖IEEE期刊和会议上发表或合著了40余篇论文，包括《IEEE/ACM网络汇刊》、《IEEE通信领域精选期刊》、ACM移动自组网与计算研讨会和IEEE INFOCOM。他目前的研究兴趣包括实时调度、分布式算法，以及面向无线通信网络的强化学习。