arXiv:2304.03947v1 [cs.IR] 8 Apr 2023

面向设备端兴趣点推荐的模型无关的去中心化协同学习

Jing Long
jing.long@uq.edu.au 昆士兰大学 澳大利亚昆士兰州布里斯班

Tong Chen
tong.chen@uq.edu.au 昆士兰大学 澳大利亚昆士兰州布里斯班

Nguyen Quoc Viet Hung
henry.nguyen@griffith.edu.au 格里菲斯大学 澳大利亚昆士兰州黄金海岸

Guandong Xu
Guandong.Xu@uts.edu.au 悉尼科技大学 澳大利亚新南威尔士州悉尼

Kai Zheng
zhengkai@uestc.edu.cn 计算机科学与工程学院和深圳高等研究院，中国电子科技大学，中国四川成都

Hongzhi Yin*
h.yin1@uq.edu.au 昆士兰大学，澳大利亚布里斯班，昆士兰州

摘要

作为基于位置的社交网络（LBSN）中不可或缺的个性化服务，下一个兴趣点（POI）推荐旨在帮助人们发现有吸引力且有趣的地点。目前，大多数兴趣点推荐系统采用传统的集中式范式，严重依赖云端，利用大量收集的用户敏感签到数据训练推荐模型。尽管近期少数研究探索了用于实现弹性和隐私保护型兴趣点推荐的端侧框架，但这些研究都假设模型具有同质性，以便进行参数/梯度聚合与协作。然而，现实世界中的用户移动设备具有各种硬件配置（例如计算资源），因而设备端模型在架构和规模上存在异质性。鉴于此，我们提出了一种新颖的端侧兴趣点推荐框架，即用于端侧兴趣点推荐的模型无关协同学习（MAC），允许用户自定义自己的模型结构（例如维度和隐藏层数量）。为应对端侧用户数据稀疏的问题，我们提出基于物理距离、类别级偏好和社交网络预先筛选协作邻居。为高效、安全地吸收上述筛选出的邻居的知识，我们采用结合互信息最大化的知识蒸馏框架。MAC中的客户端无需共享敏感模型/梯度，只需在预加载的参考数据集上共享软决策。为筛除低质量邻居，我们提出了两种采样策略：性能触发采样并采用基于相似度的采样，以加快训练过程并获得最优推荐器。此外，我们设计了两种新颖的方法，在保护用户隐私的同时生成更有效的参考数据集。在两个数据集上进行的大量实验表明，MAC 优于先进的基线方法。

CCS 概念

• **信息系统 →推荐系统**。

关键词

兴趣点推荐；去中心化协同学习；模型无关

ACM 参考格式：
Jing Long、Tong Chen、Nguyen Quoc Viet Hung、Guandong Xu、Kai Zheng 和 HongzhiYin*。2018。面向端侧兴趣点推荐的模型无关去中心化协同学习。载于论文集：请填写权利确认邮件中的正确会议名称（会议简称 *'XX*）。ACM，纽约，纽约州，美国，10 页。https://doi.org/ XXXXXXX.XXXXXXX

1 INTRODUCTION

近年来，基于位置的社交网络（例如 Weeplace 和 Foursquare）的快速发展，推动了下一兴趣点（POI）推荐的普及，为移动性预测、路线规划和基于位置的广告等多种应用提供了便利 [22]。在个性化兴趣点推荐中，近期基于注意力神经网络[25, 29, 45] 和图网络 [23, 37] 的多种先进方法，在拥有大量历史签到数据的情况下取得了高质量的推荐效果。为了支持大规模推荐服务，通常需要一台强大的云服务器来托管所有用户的数据，并对推荐系统进行训练和推理，这使得此类服务的维护在经济和生态层面都代价高昂 [28]。此外，用户将隐私视为首要考虑因素，因此越来越不愿意向服务提供商分享自己的签到轨迹，从而影响推荐质量。除此之外，在这种模式下，用户设备仅作为终端，用于

*通讯作者。

允许免费制作本作品全部或部分内容的数字或纸质副本，供个人或课堂使用，前提是不得为营利或商业利益制作或分发副本，且副本须附有此声明及首页上的完整引文。对于本作品中归 ACM 以外的其他方所有的组件，必须尊重其版权。注明出处后，允许摘录。若要以其他方式复制、再版、发布到服务器或分发至列表，则须事先获得明确许可和/或支付费用。请联系 permissions@acm.org 申请许可。会议缩写 *'XX，2018*年*6*月*3*日至*5*日，纽约州伍德斯托克© 2018 计算机协会。ACM ISBN 978-1-4503-XXXX-X/18/06...$15.00 https://doi.org/XXXXXXX.XXXXXXX

传输用户数据并显示服务器生成的推荐结果。因此，基于云的兴趣点推荐过度依赖服务器的能力和互联网连接 [33, 46], ，削弱了其韧性。

因此，近期研究转向设备端兴趣点推荐 [16, 28, 33] ，以应对集中式范式的不足。这一新范式的核心目标是在用户侧部署一个准确且相对轻量的推荐模型，从而利用本地获取的用户数据，以完全设备端的方式生成推荐。为实现这一目标，文献中一种直接的解决方案是采用模型压缩 [1, 8, 17]。在设备端兴趣点推荐的背景下，模型压缩方法旨在将复杂的兴趣点推荐系统压缩为紧凑模型，以支持设备端推理。例如， [33] 采用教师—学生蒸馏框架，将知识从功能强大的教师模型迁移至紧凑的学生模型。经过充分训练的学生模型随后会部署到移动设备上，供所有用户使用。然而，由于服务器全程参与教师模型和学生模型的训练，该框架仍然资源消耗较大。此外，所有用户都被分配相同的模型，且没有用于应对空间活动动态变化和用户兴趣多样性的更新机制，因而导致性能欠佳。

因此，近期研究更倾向于采用协同学习（CL）框架，该框架支持模型/梯度共享。因此，CL 对复杂模型设计的要求较低，也更适合兴趣点推荐任务的动态特性。在 CL 中，基于联邦学习的兴趣点推荐系统（例如 [16]）是一种极具代表性的解决方案：用户在本地训练模型，同时将所有私人数据保留在设备端。为避免本地数据的稀疏性，中央服务器会迭代地收集并聚合这些训练好的本地模型，然后将聚合后的模型重新分发给所有用户。然而，将所有本地模型聚合为单一全局模型有助于推荐系统实现泛化，但会放大其对活跃用户偏好的偏差 [34]。因此，联邦兴趣点推荐系统中进一步提出了相应的改进方法 [19, 32, 39], ，通过将相似用户分组并执行分组聚合，使学习到的设备端模型具备更强的个性化能力。

尽管与模型压缩方法相比，基于协同学习的方法在性能和灵活性方面有所提升，但在现实世界的设备端兴趣点推荐中仍存在不足。首先，联邦兴趣点推荐系统的整个学习过程中始终涉及中央服务器。由于需要在服务器上反复通过模型参数比较或聚类来识别用户组，这种依赖只会进一步加深。因此， [28] 提出的半去中心化学习范式可视为比联邦推荐系统更进一步的一种范式，能够显著降低对中央服务器的依赖。简言之，服务器只需提供预训练模型参数，并将相似用户分配到同一组中；这只是训练早期的一次性参与。之后，用户的设备端模型通过在同一用户组内交替进行本地训练和设备间通信来优化。遗憾的是，组内协作需要在用户之间交换原始模型参数， [28], 这违背了去中心化兴趣点推荐系统对隐私和通信效率的要求。其次，人们一直强烈假设用户特定知识可以通过模型参数或梯度进行聚合。然而，模型同质性假设严重损害了这类去中心化兴趣点推荐范式的实用性，因为在实践中，每个设备端模型都应具有定制结构，以满足特定的设备容量需求 [7]。随着能够提供兴趣点推荐的用户设备种类呈指数级增长，必须放宽这一假设，以允许大量结构异构的设备端模型进行联合优化。因此，仍缺乏一种有效的基于协同学习的兴趣点推荐范式，既能适配异构的端侧推荐系统，又能最大限度地降低对中央服务器的依赖，同时确保所有端侧推荐系统具有足够的表达能力，且不会造成隐私泄露或带来沉重的通信开销。

为此，我们提出了一种新颖的设备端兴趣点推荐框架，即用于设备端兴趣点推荐的模型无关协同学习（MAC）。在 MAC 中，用户可以根据设备性能定制模型配置。由于兴趣点嵌入是推荐系统内存消耗的主要来源 [24, 33], ，我们提出仅在用户设备上存储与当前位置上下文高度相关的兴趣点嵌入。MAC 中模型无关协同学习的核心方案是一种新颖的知识蒸馏（KD）机制，可促进异构推荐模型之间的知识交换。在同构场景下，模型需要共享敏感的模型参数/梯度以进行等维聚合；而 MAC 中的模型只需在所有用户共享的参考数据集上交换软决策。在知识蒸馏过程中，我们让模型通过最大化交换的软决策之间的互信息，彼此从邻居处吸收知识。一方面，利用参考数据上的软决策可以绕过去中心化模型学习中的等维约束，从而将基于协同学习的兴趣点推荐扩展至拥有异构设备和设备端模型的更多用户。另一方面，由脱敏参考数据生成的软决策不会泄露用户敏感信息；与模型聚合 [28] 方法相比，这也大幅降低了通信开销。显然，对于每位用户而言，让其模型与所有其他模型协同在计算上难以承受，也容易导致个性化不足。因此，我们提出根据物理距离、类别偏好和社交网络距离来确定通信邻居。然而，简单的初步筛选无法过滤掉无价值的客户端，因此我们进一步提出两种邻居采样策略，即性能触发采样和基于相似度的采样，以加快训练过程并获得最优推荐器。常规知识蒸馏的另一个问题是，先前工作未考虑参考数据集的来源，而是直接将整个数据集拆分以获取参考数据集[2, 41]。然而，简单地将用户历史数据视为参考数据集会带来隐私问题，并且无法适用于冷启动场景。为此，我们设计了两种新颖的方法，即变换式生成和概率式生成，以构建更实用、更有效的参考数据集，同时保护用户隐私。本研究的主要贡献如下：

- 我们提出了一种新颖的设备端兴趣点推荐框架，即用于设备端兴趣点推荐的模型无关协同学习（MAC），用户可以据此定制模型配置，并且只存储与当前上下文高度相关的兴趣点嵌入。模型无关协同学习（CL）的核心方案是一种新颖的知识蒸馏（KD）机制，可促进异构推荐模型之间的知识交换。

- 为筛除低质量邻居，我们设计了两种采样策略，即性能触发采样和基于相似度的采样，以加快训练过程并获得最优的用户专属推荐器。此外，我们提出了两种新颖的方法，即变换式生成和概率式生成，它们能够在保护用户隐私的同时构建更实用的参考数据集。

- 我们使用两个真实世界数据集在MAC进行评估，实验结果表明，该模型在隐私保护、推荐准确率以及端侧计算资源利用效率方面均具有良好的有效性和效率。

2 预备知识

本节首先介绍本文中使用的重要符号，然后形式化定义我们的核心任务。

我们分别用 U、P、C 表示用户集合 u、兴趣点集合 p 和类别集合 c。每个兴趣点 $p \in$ P 都关联一个类别标签（例如娱乐或餐厅）$c_p \in$ C 和坐标（lon_p, lat_p）。

定义 1：签到序列。用户的签到活动表示用户 $u \in$ U 在时间戳 t访问了兴趣点 $p \in$ P。按时间顺序排列用户的签到记录后，签到序列包含用户 u_i连续访问的 M_i 个兴趣点，记为 X(u_i) $=\{p_1, p_2, \ldots, p_{M_i}\}$。

定义 2：类别序列。类别序列将签到序列 X(u_i) 中的所有兴趣点替换为其关联的类别标签，记为 $\mathrm{X}^c(u_i) = \{c_{p_1}, c_{p_2}, \ldots, c_{p_{M_i}}\}$。

定义 3：区域。区域 r 本质上是一块地理片区，可为其中的兴趣点提供额外信息。本文不对预先划定的城市行政区或郊区作任何假设，而是对所有兴趣点的坐标应用 k-均值聚类 [30]，得到一组区域 R。

定义 4：地理参考数据集。区域 r 的地理参考数据集 $\mathrm{D}^g(r) = \{\mathrm{X}_v\}_{v=1}^{V_r}$ 包含 V_r 个匿名签到序列。每个区域都有专属且唯一的参考数据集，其中的签到活动仅涉及区域 r 内的兴趣点。每个用户 u 仅持有其当前所在或曾经到访区域的区域专属参考数据集。

定义 5：语义参考数据集。语义参考数据集 $\mathrm{D}^s = \{\mathrm{X}_z^c\}_{z=1}^{Z}$ 包含 Z 个匿名类别序列，涵盖所有兴趣点类别 C。该语义参考数据集为通用数据集，由所有用户共享。我们将在第 3 节详细介绍生成 D^s 和 D^g 的策略。

任务 1：去中心化下一兴趣点推荐。在MAC中，设备/用户和中央服务器的职责定义如下：

- **用户/设备**：每位用户 u_i持有自己的数据 X(u_i)、$\mathrm{X}^c(u_i)$ 以及一个通过与本地数据联合训练、并通过与其他用户协作得到增强的结构个性化模型 (·)。为节省存储空间，我们假设模型 $\phi_i(\cdot)$ 只存储用户当前或曾经到访区域内兴趣点的嵌入，记为 $r \in \mathrm{R}(u_i)$。
- **服务器**：服务器负责为所有用户识别邻居集合（使用收集到的低敏感度数据），并收集参考数据集，再将这些信息传回用户。

随后，通过促进用户之间的模型无关协作，我们旨在为每位用户学习一个性能良好的本地模型，用于估计她下一次移动可能到访的兴趣点排序列表。

3 框架

本节介绍我们设计的MAC框架，其概览如图1所示。主要组件包括：(1) 一个**本地目标函数**，用于指导每位用户设备端模型的优化。(2) 一个**邻居识别流程**，由本地设备与中央服务器协作执行，以安全地识别紧密用户群组。(3) 两种**数据生成方法**，用于构建高质量且匿名的参考数据集。(4) 一种**基于知识蒸馏的协同学习方案**，结合**邻居采样**，从而在模型无关的用户组内实现更精细的知识共享。

3.1 本地目标函数

MAC 的主要目标是增强基于用户自身签到历史进行本地训练的个性化兴趣点推荐系统。为了首先获得本地训练的推荐系统，我们定义如下本地目标函数：

$$L_{loc}(u_i) = l\left(\phi_i\left(\mathcal{X}(u_i)\right), \mathcal{Y}(u_i)\right), \tag{1}$$

其中，ϕ_i(X(u_i)) 是推荐系统 $\phi_i(\cdot)$ 根据 X(u_i) 作出的预测。在兴趣点推荐场景中，预测依次基于历史兴趣点序列 $\{p_1\}$，$\{p_1, p_2\}$ ..... $\{p_1, p_2 \quad p_{M_i-1} \quad u_i = p_2 \; p_3 \quad p_{M_i}$ ，.....} 作出，而 Y( ) {，.....，} 是相应真实兴趣点的集合。l 是用于量化预测误差的损失函数（在我们的场景中为交叉熵）。值得注意的是，MAC 与大多数以深度神经网络为基础模型的兴趣点推荐系统兼容，而我们工作的关键创新在于分布式协同学习（CL）范式，该范式与部署在设备端的各个兴趣点推荐系统协同工作。

3.2 邻居识别

为缓解数据稀缺问题（相较于在各自孤立的环境中训练单个设备端推荐系统），并减少与所有用户设备通信相比可能产生的噪声和开销，MAC 中的每个用户设备 u_i都可以与其邻居交换知识。邻居是指与 u_i具有较高亲和度的一部分用户。为实现稳健的持续学习过程，我们提出了两种并行策略，用于识别优质邻居，具体如下：u_i

地理邻居。鉴于兴趣点推荐任务对位置的敏感性，如果用户经常访问同一区域 $r \in$ R内的场所，他们就具有较高的地理亲和度 u_i，因此，地理位置相似的用户所提供的信息可相互助益，促进未来的移动预测。我们将这类用户称为 u_i的地理邻居，记作G(u_i)。如上所述，通过采用适中的R粒度，我们可以识别出相对密集的邻居集合，同时避免泄露用户隐私。

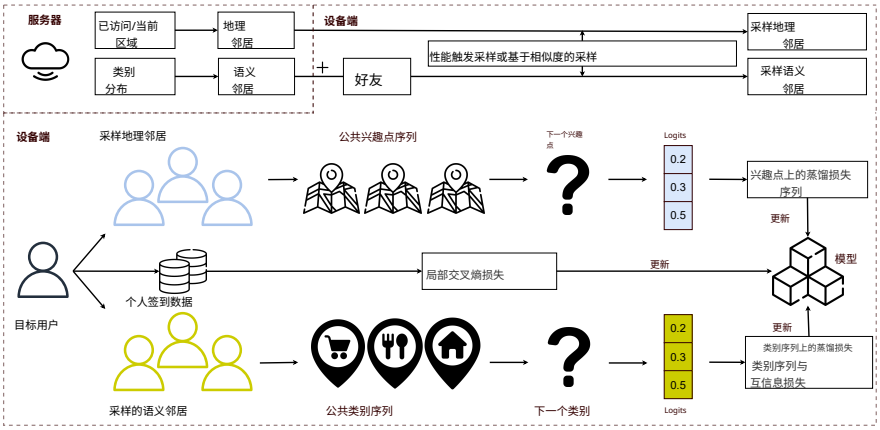

图 1：我们提出的 MAC 概览。

我们可以用$\mathrm{R}(u_i) = \{r_0^i, r_1^i, r_2^i, \ldots\ldots r_{R_i}^i\}$表示$u_i$访问过的所有区域，其中 r_0^i 表示用户当前所在的区域（对应于 p_{M_i}），其余区域则是用户曾经到访过的区域。按照我们的定义，对于两个用户 u_i和u_j，只有当u_j访问过 u_i当前所在的区域，即 $r_0^i \in \mathrm{R}(u_j)$ 时，u_j才被视为 u_i的地理邻居。请注意，这种关系具有方向性；我们只通过 r_0^i 识别u_i的邻居，而不依据所有历史到访区域，因为对 u_i的下一个兴趣点进行预测，主要依据的是她最近一次签到 p_{M_i}。

语义邻居。与此同时，用户之间的相关性不仅体现在彼此距离较近，也体现在语义层面的相似性。也就是说，即使用户在地理位置上相距甚远，只要他们访问过的兴趣点属于相同类别 [23],，他们之间也可能高度相关，我们将这类用户称为 u_i 的语义邻居，记作 $\mathrm{S}(u_i)$。基于此，我们利用类别级用户偏好来量化两个用户之间的语义相似性。形式上，我们用 $CP(u_i) = \{\mathbb{P}(c_1)\ \mathbb{P}(c_2)\ \ldots\mathbb{P}(c_{|\mathbb{C}|})\}$ 表示每个用户在所有 $|\mathbb{C}|$ 个兴趣点类别上的分布，该分布可从 $\mathrm{X}^c(u_i)$ 中轻易观察到。随后，我们采用 Kullback–Leibler（KL）散度 [13] 来量化两个用户类别偏好之间的距离：

$$d_{cat}(u_i, u_j) = KL\left(CP(u_i) \,||\, CP(u_j)\right). \tag{2}$$

对于用户 u_i，我们将距离最小的 h 个用户 d_{cat}（u_i，u_j）视为语义邻居。我们还通过将与 u_i是好友的用户加入 $\mathrm{S}(u_i)$，来考虑明确的社交关系。

值得注意的是，在去中心化设置中，用户仅凭自己的签到数据，无法获取用于确定其地理/语义邻居的信息。因此，我们会维护一个中央服务器，用于收集从 $\mathrm{X}^c(u_i)$ 计算得出的相关区域 $\mathrm{R}(u_i)$ 和类别偏好 $CP(u_i)$，从而由服务器告知所有用户其邻居 ID。与传输原始轨迹或模型相比，上传相关区域和类别偏好带来的隐私风险要低得多 [28]。

3.3 生成匿名参考数据

给定地理邻居 $\mathrm{G}(u_i)$ 和语义邻居 $\mathrm{S}(u_i)$，新的挑战是如何提取它们的知识，以增强每位用户的个性化模型。尽管模型聚合已在以往工作中证明有效 [16, 28], 但它要求所有参与方持有同构模型，而且推荐效用受计算资源最少的设备所限。因此，随着个人设备日益多样化、计算能力各不相同，参数共享变得不切实际。为此，我们提出一种新颖的、基于知识蒸馏的协同学习方法。客户端无需传输梯度或权重，只需与邻居共享其在脱敏参考数据集上的软决策。从本质上说，对于相同的参考数据集，这些生成的软决策能够简洁地表征各个设备端模型。此外，与交换梯度或模型相比，使用软决策可以显著降低通信成本，而且敏感性要低得多。

以往的持续学习框架 [2, 41]曾尝试为所有用户分配相同的参考数据集，以促进这种模型无关的知识交换，但在兴趣点推荐任务中，这种做法存在严重局限。首先，参考数据集应覆盖所有区域的兴趣点，以便所有用户生成彼此可比较的概率分布。显然，这会损害软决策的质量，因为每个用户的本地模型都是基于受地理范围限制的数据训练的，几乎无法对熟悉区域以外的兴趣点做出可靠预测。其次，存储和处理覆盖全部兴趣点的签到序列会给用户设备带来沉重负担，从而降低 MAC 的实用性。第三，尽管参考数据集在协同学习中发挥着重要作用，获取这类数据集的一种常见做法 [2, 41] 是从用户数据中抽取真实样本子集，这会损害用户隐私，进而降低其在现实场景中的可用性。

地理与语义参考数据集。 为了在地理邻居之间进行知识共享，并确保每个本地模型在参考数据集上的预测质量，我们为每个区域准备一个参考数据集，其中每个特定区域的参考数据集 $\mathrm{D}^g(r) = \{\mathrm{X}_v\}_{v=1}^{V}$ 包含 V条区域 r内的匿名签到序列。基于此，用户 u_i可以通过比较各方针对 $\mathrm{D}^g(r_0^i)$ 中每条签到序列的软决策，与她的地理邻居进行交流（请记住，r_0^i是 u_i当前所在的区域）。此外，由于现在每个 $\mathrm{D}^g(r)$ 仅包含特定区域的兴趣点，因此需要存储的兴趣点信息量大幅减少。遗憾的是，这种策略不适用于与语义邻居进行协同学习的情形，因为 u_i 及其语义邻居之间可能相距甚远，而且他们本地训练得到的

模型在同一个特定区域参考数据集上可能表现出不同的行为。因此，我们不让他们预测实际兴趣点，而是让他们在语义参考数据集 $\mathrm{D}^s = \{\mathrm{X}_z^c\}_{z=1}^{Z}$ 上预测高层类别转移；该数据集包含 Z 条匿名类别序列，其定义见第 2 节。类似地，u_i 与其语义邻居通过针对 D^s 计算软决策来实现知识共享。鉴于该数据集仅对类别转移进行建模，且兴趣点类别的数量相对较少，共享一个通用语义参考数据集即可满足预测质量和内存效率的要求。

与此同时，隐私方面的挑战仍需进一步关注。为在构建参考数据集时维护所有用户的隐私，我们采用伪数据作为一种自然解决方案。具体而言，我们提出了两种数据生成策略，即变换式生成和概率式生成，使 MAC 能够生成合理且匿名的轨迹。下面介绍这两种策略，它们在 MAC 中交替采用。

变换式生成。我们不直接发布用户的签到序列，而是提出通过一系列变换对其进行混淆；这些变换广泛用于增强时间序列数据 [11, 12, 14]。简而言之，我们通过分解-重组流程，将两位用户的签到序列/类别序列拆分后重新组合，生成经过改造的序列；这些序列虽与原始序列不同，却模仿了原始序列。由于此过程中会交换原始签到记录，因此仅当两位用户 u_i 和 u_m 满足以下条件时，我们才允许进行此操作：(1) 两人互为好友；(2) 两人至少到访过同一个兴趣点。当 u_m 向 u_i 发送其签到序列后，即可判断是否满足这一条件。具体而言，对于两个签到序列 $\mathrm{X}(u_i)$、$\mathrm{X}(u_j)$，我们从一个共同兴趣点 $p^{ij} \in \mathrm{X}(u_i) \cap \mathrm{X}(u_j)$ 处切分，将每个序列分解为两个子序列，即 $\mathrm{X}'(u_i) = \{p_1^i, ..., p^{ij}\}$ 和 $\mathrm{X}''(u_i) = \{p^{ij}, ..., p_{M_i}^i\}$（对应 $\mathrm{X}(u_i)$），以及 $\mathrm{X}'(u_j) = \{p_1^j, ..., p^{ij}\}$ 和 $\mathrm{X}''(u_j) = \{p^{ij}, ..., p_{M_j}^j\}$（对应 $\mathrm{X}(u_j)$）。随后，我们将 $\mathrm{X}'(u_i)$ 与 $\mathrm{X}''(u_j)$ 拼接，并将 $\mathrm{X}'(u_j)$ 与 $\mathrm{X}''(u_i)$ 拼接，形成两个新序列 X_1^g 和 X_2^g，并从两个序列中删除重复的 p^{ij}。对于这两个新生成的序列，我们将其区域 r 指定为访问频率最高的区域，并剔除区域 r 之外的兴趣点，以确保生成的所有兴趣点序列都针对特定区域。类似地，可以通过对用户的类别序列进行变换式生成，构建通用语义参考数据集 D^s；这一过程无需额外进行特定区域处理。

概率式生成。虽然变换式生成仅使用扰动后的数据，但这些数据源自用户的原始历史数据。为满足更严格的隐私限制，并应对签到记录有限的区域，我们提出一种替代方案：仅利用用户不敏感类别序列的统计信息生成数据，我们称之为概率式生成。首先，我们根据所有用户提交的类别序列计算所有类别的条件概率。形式上，我们用 $P(c_n) = \{\mathbb{P}(c_n|c_1)\ \mathbb{P}(c_n|c_2)..., \mathbb{P}(c_n|c_m)..., \mathbb{P}(c_n|c_{|C|})\}$ 表示 c_n 的所有条件概率，其中每个 $\mathbb{P}(c_n|c_m)$ 的计算方式如下：

$$\mathbb{P}(c_n|c_m) = \frac{\mu(c_n|c_m)}{\sum_{m'=1}^{|C|} \mu(c_n|c_{m'})}, \tag{3}$$

其中，$\mu(c_n|c_m)$ 表示 c_m 紧接在 c_n 之后出现的次数。随后，可以生成伪类别序列 X^s

具体方法为：(1) 随机选择一个类别作为起点；然后 (2) 根据其前一个类别的条件概率，反复确定后续类别。通过迭代生成多个 X^c，我们可以创建一个覆盖所有类别的语义参考数据集 D^s。下一步是基于该语义参考数据集生成 POI 序列。在每个区域内，我们随机采样一条与类别序列 $\mathrm{X}^c \in \mathrm{D}^s$ 对应的兴趣点序列，并进一步限制任意两个连续兴趣点之间的距离小于 5km。

3.4 通过知识蒸馏进行协作

参考数据集准备就绪后，我们设计了一种新颖的、基于知识蒸馏的协同学习协议，以从地理邻居 $\mathrm{G}(u_i)$ 和语义邻居 $\mathrm{S}(u_i)$ 提供的软决策中提取知识。

与地理邻居进行协同学习。给定用户 u_i 及其当前区域的地理参考数据集 $\mathrm{D}^g(r_0^i)$，通过最小化其与地理邻居 $\mathrm{G}(u_i)$ 在 $\mathrm{D}^g(r_0^i)$ 上软决策的分歧来实现与地理邻居的协同学习，分歧量化如下：

$$L_{geo} = \frac{1}{|\mathcal{G}(u_i)|} \sum_{u_j \in \mathcal{G}(u_i)} \left(\sum_{\mathcal{X} \in \mathcal{D}^g(r_0^i)} \left\| \phi_i(\mathcal{X}) - \phi_j(\mathcal{X}) \right\|_2^2 \right), \tag{4}$$

其中，$\phi_i(\cdot)$ 和 $\phi_j(\cdot)$ 分别表示 u_i 及其邻居所拥有的本地推荐系统。

通过语义邻居进行协同学习。类似地，我们也可以对齐 u_i 与 $\mathrm{S}(u_i)$ 在语义参考数据集 D^s 上的软决策。然而，由于参考数据集中仅记录了类别标签，因此只能根据类别偏好进行预测，我们无法直接通过比较用户的软决策来更新 $\phi_i(\cdot)$ 中的兴趣点嵌入。为此，我们设计了一种两步策略，从语义邻居处获取知识。首先，我们通过以下方式更新 $\phi_i(\cdot)$ 中的类别嵌入：

$$L_{cat} = \frac{1}{|\mathcal{S}(u_i)|} \sum_{u_j \in \mathcal{S}(u_i)} \left(\sum_{\mathcal{X}^c \in \mathcal{D}^s} \left\| \psi_i(\mathcal{X}^c) - \psi_j(\mathcal{X}^c) \right\|_2^2 \right), \tag{5}$$

其中，$\psi_i(\cdot)$ 是一个轻量级预测器，用于根据历史类别轨迹 $X^c \in \mathrm{D}^s$ 推断所有类别上的概率分布。因此，由于兴趣点 p 的嵌入及其关联类别 c_p 的嵌入可视为该兴趣点的两个视图，我们进一步通过最大化这两个视图之间的互信息（MI）来更新兴趣点嵌入：

$$L_{MI} = -\sum_{\forall p} \log \frac{\exp(f(p, c_p))}{\sum_{\forall c' \neq c_p} \exp(f(p, c'))}, \tag{6}$$

其中 $f(\cdot)$ 是POI 与类别之间成对相似度的函数。在MAC 中，它由一个双线性网络实现，该网络以 POI 和类别嵌入作为输入：

$$f(p, c) = \sigma(\mathbf{e}_p^\top \mathbf{W} \mathbf{e}_c), \tag{7}$$

其中，$\mathbf{e}_p \in \mathbb{R}^d$ 和 $\mathbf{e}_c \in \mathbb{R}^d$ 分别是兴趣点和类别嵌入，$\mathbf{W} \in \mathbb{R}^{d \times d}$ 是可学习权重矩阵。通过优化损失函数，我们可以最大化正样本对之间的互信息，同时最小化负样本对之间的互信息。然后，

算法1 优化 MAC。除非另有说明，所有流程均在用户端实现。

/*服务器端参与开始*/
1：服务器从所有 $u_i \in U$接收 $R(u_i)$、 $CP(u_i)$;
2：针对所有 $u_i \in$，根据 $R(u_i)$和 $CP(u_i)$生成G(u_i)和S(u_i) $\mathcal{U}$;
3：为所有 $r \in$R 构造 D$^g(r)$ 和 D^{sem}，采用变换生成或概率式生成;
4：将 G(u_i)、S(u_i)、D$^g(r_0^i)$、D^{s}发送给每个 $u_i \in U$;
/*服务器参与结束*/
5: **对$\mathcal{U}$ 并行执行**
6: 使用 u_i 的社交好友填补 S(u_i);
7: 为所有 $u_i \in$U 初始化 $\phi_i(\cdot)$、G$_\prime(u_i)$、S$_\prime(u_i)$;
8: **重复**
9: 分别接收 D$^g(r_0^i)$、D^s 上的软决策来自 G$_\prime(u_i)$、S$_\prime(u_i)$;
10: 接收来自 N^s 的 D^c 上的软决策 $_{cat}(u_i)$;
11: 针对 $L_{loc}+\gamma(\mu L_{geo}+(1-\mu)L_{sem})$ 执行一个梯度步用于更新 $\phi_i(\cdot)$;
12: 使用性能触发或基于相似度的采样;
13: **直到** 收敛
14: **结束循环**。

基于语义邻居进行协同学习的最终损失定义为这两部分的组合：

$$L_{sem} = L_{cat} + L_{MI}. \tag{8}$$

3.5 使用动态邻居采样学习本地推荐器

MAC 的优化流程如算法1所示。首先，我们在服务器端执行邻居识别，并准备好所有参考数据集（第1–3行）。随后，服务器将相应的参考数据集和邻居信息发送给所有用户（第4行），至此服务器端的工作结束。在设备端（第5–14行），每个用户都会基于协同损失 $L_{loc}+\gamma(\mu L_{geo}+(1-\mu)L_{sem})$ 迭代更新其本地推荐系统，其中 γ 和 μ 共同控制不同知识蒸馏组件的作用强度。

同时，尽管邻居识别过程为 u_i提供了大量可能有助于学习 $\phi_i(\cdot)$ 的用户，但 G(u_i)/S(u_i) 中并非所有用户都具有价值，反而可能降低本地模型的质量。除了噪声的影响之外，在每个优化步骤中与所有预定义邻居通信也效率低下。因此，我们进一步提出两种邻居采样策略，从 α 用户中抽取一个较小的子集 G$_\prime(u_i)$ $\in$G(u_i) 和 S$_\prime(u_i)$ $\in$S(u_i)，用于协同学习。将式（4）和式（5）中的完整邻居集合 G(u_i) 和 S(u_i) 替换为 G$_\prime(u_i)$ 和 S$_\prime(u_i)$，即可加快训练过程并获得最优推荐器。

性能触发采样。直观而言，选择能够提供高质量知识的邻居是有益的，而本地损失函数 $L_{loc}(u_i)$ 的波动是判断从 u_i 的邻居那里蒸馏出的知识是否足够有用的重要指标。因此，在每个协同学习轮次 o 中，如果 $\Delta L^o_{loc}(u_i)$ 的变化低于预定义阈值 τ，则此前采样的邻居 G$_\prime(u_i)$/S$_\prime(u_i)$ 将从 G(u_i)/α 中重新抽取。在我们的工作中，我们将 $\Delta L_{loc}(u_i)$ 定义为以下比率形式：

$$\Delta L^o_{loc}(u_i) = \frac{|L^o_{loc}(u_i) - L^{o-1}_{loc}(u_i)|}{L^{o-1}_{loc}(u_i)} \times 100\%, \tag{9}$$

其中，我们可以调整 τ 以主动控制对性能变化的敏感度，并在本文中对两种邻居类型均采用 τ = 1%。请注意，为避免重复，我们以均匀随机且有放回的方式抽取样本。

基于相似度的采样。自然地，另一种可行的解决方案是通过相似度度量，动态缩小在 G(u_i)/S(u_i) 中进行通信的邻居范围。为此，我们通过在每次协同学习迭代中比较参考数据集，利用关于参考数据集的软决策，并通过 KL散度 [13]:

$$d_{soft}(u_i, u_j) = \sum_{\forall X \in \mathcal{D}^{ref}} KL\left(\phi_i(X) \,||\, \phi_j(X)\right), \tag{10}$$

其中，当 $u_j \in$G$_\prime(u_i)$ 时，D^{ref} =D$^g(r_0^i)$；当 $u_j \in$S$_\prime(u_i)$ 时，D^{ref} = D^s。然后，对于用户 u_i，分别从 G(u_i) 和 S(u_i) 中选取具有最小 $d_{soft}(u_i, u_j)$ 的 β用户。

在 MAC 中，我们采用基于相似度的采样或性能触发采样，因为二者可以相互替代。此外，由于所有模型都在随时间变化，采样过程会动态调整 G$_\prime(u_i)$ 和 S$_\prime(u_i)$，如算法1第 12 行所示。

4 实验

本节中，我们在两个真实世界数据集上进行了大量实验，以评估 MAC 的有效性和效率。

表 1：数据集统计信息。

	Foursquare	Weeplace
#用户	7,507	4,560
#兴趣点	80,962	44,194
#类别	436	625
#签到记录	1,214,631	923,600
#每位用户的签到次数	161.80	202.54

4.1 数据集和评估协议

我们采用两个真实世界数据集来评估所提出的 MAC，即 Foursquare [9] 和 Weeplace [27]。这两个数据集均包含用户在纽约、洛杉矶和芝加哥的签到历史。按照 [3, 22], , 我们移除了交互次数少于 10 次的用户和兴趣点。表 1 总结了这两个数据集的统计信息。在这些数据中，我们随机选取 10% 的签到序列作为参考数据集，并确保覆盖所有兴趣点。基于所选的签到活动，我们还采用两种策略生成参考数据集。随后，受 [35, 36],启发，我们采用留一法协议进行评估。具体而言，对于其余每个签到序列，最后一次签到对应的兴趣点用于测试，倒数第二次签到对应的兴趣点用于验证，其余所有兴趣点用于训练。此外，最大序列长度设为 200。对于每个真实值，我们不对所有电子商务商品进行排序 [21], , 而是仅将其与同一序列所在区域内 200 个未访问过且最近的兴趣点配对，作为排序候选项。其原因是，与电子商务商品不同 [21],，在兴趣点推荐场景中

表 2：与基线方法的推荐性能比较。

	Foursquare				Weeplace			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
MF	0.0835	0.0598	0.0962	0.0669	0.1045	0.0612	0.1293	0.0872
LSTM	0.1920	0.1227	0.2812	0.1648	0.2215	0.1356	0.3301	0.1578
STAN	0.2914	0.1756	0.4157	0.2524	0.3205	0.1846	0.4503	0.2783
LLRec	0.2771	0.1488	0.3406	0.1908	0.2912	0.1781	0.3613	0.2255
PREFER	0.2895	0.1696	0.3607	0.2183	0.3055	0.1874	0.3707	0.2359
DCLR	0.3078	0.1797	0.4297	0.2653	0.3304	0.1912	0.4664	0.2838
D-Dist	0.2257	0.1193	0.2915	0.1621	0.2474	0.1343	0.3581	0.1933
SQMD	0.2818	0.1471	0.4054	0.2357	0.2953	0.1531	0.4342	0.2360
MAC-P	**0.3138**	**0.1865**	**0.4319**	**0.2712**	**0.3414**	**0.2032**	**0.4843**	**0.2937**
MAC-S	0.3055	0.1843	0.4257	0.2637	0.3365	0.1988	0.4734	0.2845

兴趣点推荐具有位置敏感性，用户很少会连续前往彼此相距很远的两个兴趣点 [23, 28]。基于此，我们采用两种排序指标，即排名k处的命中率（HR@k）和排名k处的归一化折损累积增益（NDCG@k） [38]。其中，HR@k仅衡量真实值出现在前k个候选项列表中的次数，而 NDCG@k则关注真实值能否尽可能排在靠前位置。

4.2 基线方法与实验设置

我们将 MAC 与集中式和设备端兴趣点推荐系统进行比较：

- **MF** [26]：这是一个基于用户-物品矩阵分解的经典集中式兴趣点推荐系统。
- **LSTM** [18]：这种循环神经网络能够捕捉序列数据中的短期和长期依赖关系。
- **STAN** [29]：它通过双重注意力方法学习签到轨迹中的显式时空相关性。
- **LLRec** [33]：它采用师生训练策略来获得可在本地部署的压缩模型。
- **PREFER** [16]：这种联邦兴趣点推荐范式允许服务器收集并聚合本地训练的模型，同时重新分发联邦模型。
- **DCLR** [28]：该去中心化协同学习框架通过模型聚合，使本地训练的模型能够在同质邻居之间共享知识。
- **D-Dist** [2]：该方法旨在让本地训练的模型能够与随机的异质邻居进行协作，方法是比较它们在公开参考数据集上的软决策。
- **SQMD** [41]：这也是一种去中心化蒸馏框架，通过比较邻居在共享参考数据集上的响应来定义邻居关系。

4.3 实验设置

如上所述，本研究几乎兼容任何集中式兴趣点推荐模型。为实现先进的准确率，我们采用 STAN [29] 作为基础模型。此外，由于本研究支持异构结构，我们将潜在维度 $d \in \{8, 16, 32, 64, 128\}$ 随机分配给用户，每种维度各占 20%。这一设置也应用于其他异构框架（即 D-Dist 和 SQMD）。为确保公平，所有其他基线方法均分别使用上述维度进行评估，并对最终结果取平均。此外，在本研究中，每个城市被划分为 5 个区域，划分方式是采用第 2 节所述的 K-means 聚类。值得注意的是，在 MAC 中，我们实现了性能触发和基于相似度的方法两种邻居采样方式，分别记为 MAC-P 和 MAC-S。

对于超参数，我们将 α 设为 5、β 设为 10、γ 设为 0.5，μ 设为 0.7。上述四个超参数的影响将在第 4.7 节中进一步讨论。此外，我们将 h 设为 50，学习率设为 0.002，所有深层网络的丢弃率设为 0.2，批量大小设为 16，最大训练轮次设为 50。

4.4 推荐效果

所有兴趣点推荐系统的性能比较总结于表 2 中，我们观察到以下发现。

在集中式兴趣点推荐器中，长短期记忆网络在两个数据集上的表现均优于矩阵分解，这是因为它能有效利用签到活动序列中的短期和长期依赖关系。此外，得益于连续和非连续签到活动之间的时空相关性，STAN 的准确率高于长短期记忆网络。与先进的基线模型 STAN 相比，我们的方法仍能取得极具竞争力的结果。这是因为集中式 STAN 模型使用多个城市的签到记录进行训练，而从一个城市学到的知识对于其他城市的推荐任务可能会产生噪声，从而导致 STAN 的性能较差。相反，我们的工作能够实现更好的个性化，并借助协同学习架构学习表达能力更强的模型。

与此同时，MAC 在两个数据集的所有指标上均优于所有设备端兴趣点推荐器。具体而言，LLRec 的表现最差，因为训练过程中未纳入所有个人数据，忽略了个人偏好；而我们的方法只需承担可忽略的隐私风险，就能大幅提升准确率。尽管 PREFER 显著提高了准确率，但仍需收集并汇总用户的个性化模型，使用户隐私面临潜在泄露风险。因此，MAC 更能以较少依赖云端的方式，同时提供更准确的推荐和更强的隐私保护。

4.5 效率分析

在所有同构端侧推荐系统中，DCLR 凭借有效的协同学习策略取得了最佳性能。然而，它受制于一项限制：所有参与者都必须持有同构模型才能进行模型聚合。在

表 3：去中心化兴趣点推荐器的模型大小（kb）和准确率。

	d = 8		d = 16		d = 32		d = 64		d = 128	
	大小	HR@10	大小	HR@10	大小	HR@10	大小	HR@10	大小	HR@10
DCLR	1130	0.3674	2252	0.4413	4502	0.5417	9020	0.5165	18127	0.4651
D-Dist	1057	0.2863	1963	0.3309	3852	0.3514	7968	0.3925	15564	0.3293
SQMD	1157	0.3637	2054	0.4194	3648	0.4845	8245	0.4585	16362	0.4449
MAC-P	232	0.4057	462	0.4678	694	0.5404	1316	0.5114	3186	0.4963
MAC-S	226	0.4026	479	0.4341	742	0.5416	1276	0.5010	2987	0.4877

表 4：训练时间（小时）。

	Foursquare	Weeplace
DCLR	76	57
D-Dist	24	19
SQMD	51	39
MAC-P	40	32
MAC-S	29	21

相比之下，MAC 不仅能像 D-Dist 和 SQMD 一样支持用户使用不同的维度，还能让用户设备通过存储部分兴趣点嵌入，有效利用有限的存储资源。为验证上述观点，我们针对所有异构端侧推荐系统，记录了潜在维度为 $d \in \{8、16、32、64、128\}$ 时的平均模型大小（kb）和推荐准确率（Weeplace 上的 HR@10）。结果如表 3 所示。可以看到，在所有维度下，MAC 的平均模型大小都远小于其他三种端侧推荐系统。此外，在所有维度下，MAC 的推荐准确率仍优于 DCLR 和其他异构模型，这表明 MAC 能够更高效地利用设备端资源，提供性能更出色的推荐。

此外，我们提出的邻居识别和邻居采样策略可以加快训练过程。我们记录模型收敛前的训练时间，结果如表 4 所示。可以观察到，D-Dist 的训练时间最短，因为网络中的每台设备在每次迭代中都会与一组固定的对等节点通信。作为交换，D-Dist 的推荐准确率最差。除此之外，MAC 的收敛速度快于所有其他模型，同时能够提供更准确的推荐。有趣的是，与 MAC-P 相比，MAC-S 的推荐准确率更高，但它需要更长的训练时间，因为它倾向于更频繁地改变邻居集合构成。相反，MAC-S 仅与相对稳定的邻居集合通信，因此收敛更快，但容易陷入局部最优解。

4.6 参考数据生成分析

为验证第 3.3 节所述两种参考数据集生成策略的有效性，我们评估了 MAC-P 和 MAC-S 在原始参考数据集、变换生成的参考数据集以及概率生成的参考数据集上的推荐准确率。结果如表 5 所示。显然，原始参考数据集上的性能与变换生成的参考数据集上的性能相近。这证明了变换策略在提供强隐私保护的同时仍然有效。与此同时，尽管概率生成的参考数据集上的性能略有下降，但对于历史签到活动不足的区域，该方法仍然适用。请注意，其他所有实验均基于变换生成的参考数据集。

4.7 超参数敏感性

本节展示四个超参数对 MAC 性能的影响。具体而言，我们评估性能触发采样中的样本量 α 对 MAC-P 准确率和训练时间的影响，以及基于相似度的采样中的样本量 β 对

表 5：参考数据集来源对推荐准确率的影响（展示 HR@10）。

	Foursquare		Weeplace	
	MAC-P	MAC-S	MAC-P	MAC-S
原始	0.4357	0.4232	0.4901	0.4698
变换生成	0.4319	0.4257	0.4843	0.4734
概率生成	0.4173	0.4129	0.4715	0.4581

MAC-S 准确率和训练时间的影响、控制来自邻居的知识的权重 γ 对 MAC-P 和 MAC-S 准确率的影响，以及用于通信的地理邻居与语义邻居之间的权衡 μ 对 MAC-P 和 MAC-S 准确率的影响。结果如图 2 所示。

影响 α。我们在一系列 $\alpha \in \{1、5、10、20、40\}$ 上进行了实验。总体而言，较大的 α 会损害 MAC-P 在两个数据集上的性能，因为其中包含了低质量邻居，但较低的通信频率能减少训练时间。在极端情况下，与所有预定义邻居进行通信（虚线），我们记录到最低准确率，这证明了性能触发采样的有效性。

影响 β。我们研究了 $\beta \in \{1、5、10、20、40\}$ 的影响。随着 β 从 1 增加到 10，MAC-S 的准确率总体呈上升趋势，因为获得了更有价值的知识。然而，当 β 超过 10 时，由于纳入了低质量邻居，准确率开始下降。同时，随着处理的邻居增多，训练时间也持续增加。与极端情况（虚线）相比，MAC-S 以更短的训练时间取得了更高的准确率，体现了基于相似度的采样的有效性。

影响 γ。我们在 {0、0.1、0.3、0.5、0.7、0.9、1} 下评估了 γ。显然，如果用户未能与邻居共享知识（$\gamma = 0$），准确率最低，这表明协同学习框架的重要性。然而，如果来自邻居的知识所占比例过高（$\gamma > 0.5$），性能就会下降。

对 μ**的影响** μ 在 $\{0,0.1,0.3,0.5,0.7,0.9,1\}$中进行了研究。在大多数情况下，当 $\mu = 0.7$ 时，性能最佳，这表明与语义因素相比，地理因素更为重要。

5 相关工作

本节回顾相关领域的最新文献，包括用于兴趣点推荐的集中式模型、用于兴趣点推荐的设备端框架，以及分布式知识蒸馏。

5.1 NextPOI 推荐

为了帮助人们通过分析用户与兴趣点之间的交互来发现有吸引力的地点，早期模型主要侧重于矩阵分解 [26] 和马尔可夫链 [10, 48]。近期，基于循环

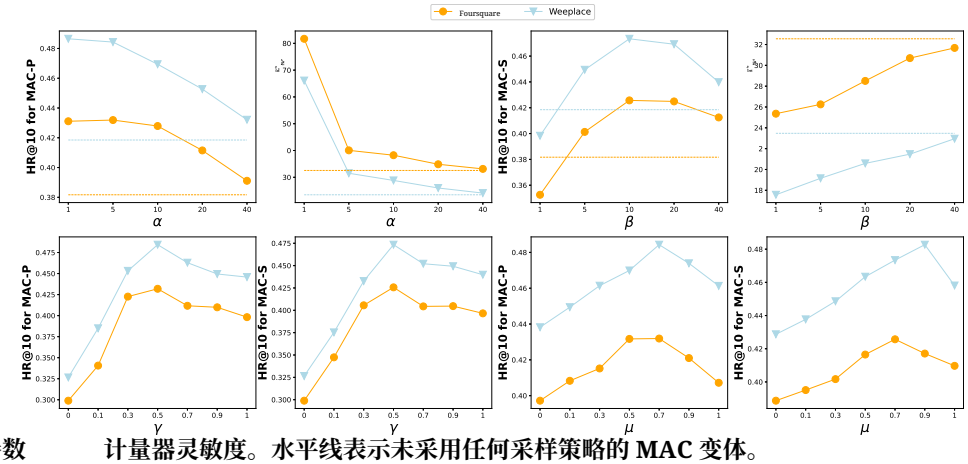

图 2：超参数 计量器灵敏度。水平线表示未采用任何采样策略的 MAC 变体。

神经网络（RNN）的模型已被证明能够有效捕捉兴趣点序列中的时空依赖关系 [5, 6, 20, 42, 44]。与此同时，SGRec [23] 构建图增强的兴趣点序列，以充分捕捉语义相关兴趣点中的协同信号并挖掘序列属性，其准确率优于基于RNN的模型。此外，注意力神经网络 [4, 29, 40, 43] 利用自注意力层捕捉序列中所有签到活动的相对时空信息。值得注意的是，以上所有模型都基于云端，因此不可避免地带来隐私问题，并且对强大的云端算力需求很高。相比之下，MAC采用分布式学习范式，提供安全稳定的服务。

5.2 设备端兴趣点推荐

事实证明，端侧框架能够有效解决基于云的兴趣点推荐学习中的大多数缺陷。直观而言，Wang 等人 [33] 在移动设备上部署了压缩模型，以提供安全、稳定的兴趣点推荐。为保持整个端侧框架的鲁棒性，本地压缩模型继承了使用公开数据训练的教师模型的知识。随后，Guo 等人 [16] 提出了一种用于 POI 推荐的联邦学习框架，使边缘服务器能够收集并聚合本地训练的模型，再将聚合后的模型发送给所有用户。显然，上述模型仍然高度依赖云服务器。在此基础上，Jing 等人 [28] 提出了一种结合设备协作的半去中心化学习范式，使用户设备能够从两类邻居那里学习并整合知识。然而，所有上述基于协同学习的兴趣点推荐系统都基于一个强假设：所有端侧模型必须采用相同设计，才能通过参数/梯度聚合实现用户特定知识共享。相比之下，MAC 支持模型结构异构。

5.3 分布式知识蒸馏

知识蒸馏最初是在模型压缩领域提出的[31],，旨在从使用大量数据训练的强大教师模型中提取知识，以提升轻量级学生模型的性能。知识通过最小化教师模型与学生模型在无标签参考数据集上的差异来传递。然而，这种知识传递方式并不局限于师生模式，其中 [15] 和 [47] 已证明，无需教师模型也能利用知识蒸馏联合训练学生模型。这也适用于异构设备网络之间的知识共享，在这种网络中，用户设备可以采用不同的模型架构 [2, 41]。我们的工作首次利用知识蒸馏，在异构POI推荐模型之间实现知识共享。

6 结论

为了支持去中心化兴趣点推荐中的异构架构，使用户能够根据设备容量定制结构，我们将互信息最大化融入知识蒸馏，以在由物理距离、类别分布和社交网络确定的基于相似度的群组之间交换信息。为剔除低质量邻居，在训练过程中，我们设计了两种策略，根据邻居的表现或其与目标用户的相似度进一步筛选邻居，从而缩短训练时间并获得最优推荐系统。此外，我们提出了两种新颖的方法，在提供强隐私保护的同时构建更实用的参考数据集。实验结果充分验证了 MAC 的有效性，表明其能够为去中心化兴趣点推荐提供优质服务。

致谢

本研究得到澳大利亚研究理事会以下项目的资助：未来学者奖（Grant No. FT210100624）、探索项目（Grant No. DP190101985）、探索早期职业研究者奖（Grant No. DE200101465 和 No. DE230101033），以及工业转型培训中心（No. IC200100022）。此外，本研究还得到国家自然科学基金委员会（No. 61972069、61836007 和 61832017）、深圳市科技研发资金基础研究项目（ICYJ 20210324133607021）以及衢州市人民政府（Grant No. 2022D037）的支持。

参考文献

[1] Y. Bengio、Nicholas Leonard 和 A. Courville。2013年。通过随机神经元进行条件计算时估计或传播梯度。计算机科学（2013年）。[2] I. Bistritz、A. Mann 和 N. Bambos。2020年。面向设备端学习的分布式蒸馏。载于《神经信息处理系统》。[3] B. Chang、Y. Park、D. Park、S. Kim 和 J. Kang。2018。面向连续兴趣点推荐的内容感知层次化兴趣点嵌入模型。载于第二十七届国际人工智能联合会议 IJCAI-18。[4] Tong Chen、Hongzhi Yin、Hongxu Chen、Rui Yan、Quoc Viet Hung Nguyen 和 Xue Li。2019年。Air：注意力意图感知推荐系统。载于2019年IEEE第35届国际数据工程会议（ICDE）。IEEE，304–315。[5] Tong Chen、Hongzhi Yin、Quoc Viet Hung Nguyen、Wen-Chih Peng、Xue Li 和 Xiaofang Zhou。2020年。面向时间预测分析的序列感知因子分解机。载于2020年IEEE第36届国际数据工程会议（ICDE）。IEEE，1405–1416。[6] Tong Chen、Hongzhi Yin、Quoc Viet Hung Nguyen、Guanhua Ye、Zi Huang、Yang Wang 和 Meng Wang。2020年。试试这个：个性化且可解释的替代品推荐。载于第43届ACM SIGIR国际会议论文集：信息检索研究与发展。891–900。[7] T. Chen、H. Yin、Y. Zheng、Z. Huang、Y. Wang 和 M. Wang。2021年。学习弹性嵌入以定制设备端推荐系统。[8] W. Chen、J. T. Wilson、S. Tyree、K. Q. Weinberger 和 Y. Chen。2015年。利用哈希技巧压缩神经网络。[9] Z. Chen、H. Cao、H. Wang、F. Xu 和 Y. Li。2020年。你还会回来或再次签到吗？了解促成城市重访和再次签到的特征。《ACM交互式移动、可穿戴与泛在技术汇刊》（2020年）。[10] C. Cheng、H. Yang、M. R. Lyu 和 I. King。2013年。下一个想去哪里：连续兴趣点推荐。载于国际人工智能联合会议。[11] H. I. Fawaz、G. Forestier、J. Weber、L. Idoumghar 和 P. A. Muller。2018年。使用合成数据进行数据增强，以借助深度残差网络实现时间序列分类。（2018年）。[12] Anibal Flores、Hugo Tito-Chura 和 Honorio Apaza-Alanoca。2021年。利用深度学习对短期时间序列预测进行数据增强。（2021年）。[13] A. Giantomassi、F. Ferracuti、S. Iarlori、G. Ippoliti 和 S. Longhi。2014年。基于定子电流测量，利用核密度估计和Kullback-Leibler散度进行电机故障检测与诊断。《IEEE工业电子学汇刊》（2014年）。[14] A. L. Guennec、S. Malinowski 和 R. Tavenard。2016年。使用卷积神经网络进行时间序列分类的数据增强。载于《欧洲数据挖掘与知识发现原理会议》。[15] Q. Guo、X. Wang、Y. Wu、Z. Yu 和 P. Luo。2020年。通过协同学习实现在线知识蒸馏。载于2020年IEEE/CVF计算机视觉与模式识别会议（CVPR）。[16] Y. Guo、F. Liu、Z. Cai、H. Zeng 和 N. Xiao。2021年。PREFER：通过联邦边缘学习实现高效且隐私保护的兴趣点推荐。《ACM交互式移动、可穿戴与泛在技术汇刊》5卷1期（2021年），1–25。[17] Geoffrey Hinton、Oriol Vinyals 和 Jeff Dean。2015年。神经网络中的知识蒸馏。《计算机科学》14卷7期（2015年），38–39。[18] S. Hochreiter 和 J. Schmidhuber。1997年。长短期记忆网络。《神经计算》9卷8期（1997年），1735–1780。[19] Mubashir Imran、Hongzhi Yin、Tong Chen、Quoc Viet Hung Nguyen、Alexander Zhou 和 Kai Zheng。2023年。ReFRS：面向动态且多样化用户偏好的资源高效联邦推荐系统。《ACM信息系统汇刊》41卷3期（2023年），1–30。[20] F. Jie、L. Yong、Z. Chao、F. Sun 和 D. Jin。2018年。DeepMove：利用注意力循环网络预测人类移动性。载于2018年万维网会议。[21] W. Krichene 和 S. Rendle。2020年。关于物品推荐的采样指标。载于KDD '20：第26届ACM SIGKDD知识发现与数据挖掘会议。[22] Ranzhen Li、Yanyan Shen 和 Yanmin Zhu。2018年。结合时间与多层级上下文注意力的下一兴趣点推荐。2018年IEEE国际数据挖掘会议（ICDM）（2018年），1110–1115。[23] Yang Li、Tong Chen、Yadan Luo、Hongzhi Yin 和 Zi Huang。2021年。通过迭代式Seq2Graph增强发现用于下一兴趣点推荐的协同信号。载于IJCAI。1491–1497。[24] Y. Li、T. Chen、P. F. Zhang 和 H. Yin。2021年。轻量级自注意力序列推荐。（2021年）。[25] D. Lian、Y. Wu、Y. Ge、X. Xie 和 E. Chen。2020年。地理感知序列位置推荐。载于KDD '20：第26届ACM SIGKDD知识发现与数据挖掘会议。[26] D. Lian、C. Zhao、X. Xie、G. Sun、E. Chen 和 Y. Rui。2014年。GeoMF：面向兴趣点推荐的联合地理建模与矩阵分解。ACM.[27] X. Liu、Y. Liu、K. Aberer 和 C. Miao。2013年。通过挖掘用户偏好转移进行个性化兴趣点推荐。载于 ACM 国际会议：信息与知识管理会议。[28] J. Long、T. Chen、Nq Viet Hung 和 H. Yin。2022年。面向下一兴趣点推荐的去中心化协同学习框架。TOIS（2022年）。[29] Y. Luo、Q. Liu 和 Z. Liu。2021年。STAN：用于下一位置推荐的时空注意力网络。[30] J. Macqueen。1967年。多变量观测分类与分析的一些方法。数学统计与概率研讨会论文集，第5届，第1卷（1967年）。[31] M. Phuong 和 C. H. Lampert。2019年。探索对知识蒸馏的理解。载于国际机器学习会议。[32] J. Rao、S. Gao、M. Li 和 Q. Huang。2021年。一种利用去中心化协同机器学习进行位置推荐的隐私保护框架。GIS汇刊（2021年）。[33] Q. Wang、H. Yin、T. Chen、Z. Huang 和 Nqv Hung。2020年。面向资源受限移动设备的下一兴趣点推荐。载于 WWW '20：2020年万维网大会。[34] Qinyong Wang、Hongzhi Yin、Tong Chen、Junliang Yu、Alexander Zhou 和 Xiangliang Zhang。2021年。快速适应且保护隐私的联邦推荐系统。（2021年）。[35] Q. Wang、H. Yin、Z. Hu、D. Lian、H. Wang 和 Z. Huang。2018年。采用对抗训练的神经记忆流式推荐网络。（2018年），2467–2475。[36] Q. Wang、H. Yin、H. Wang、Qvh Nguyen 和 L. Cui。2019年。通过生成式增强提升协同过滤。载于第25届ACM SIGKDD国际会议。[37] Zhaobo Wang、Yanmin Zhu、Qiaomei Zhang、Haobing Liu、Chunyang Wang 和 Tong Liu。2021年。用于下一兴趣点推荐的图增强时空网络。ACM数据知识发现汇刊（TKDD）（2021年）。[38] M. Weimer、A. Karatzoglou、Q. V. Le 和 A. J. Smola。2007年。CoFiRank——用于协同排序的最大间隔矩阵分解。载于神经信息处理系统。[39] A Xw、A Mn、A Jc、B Lc 和 C Kl。2020年。一种基于群体偏好的隐私保护兴趣点推荐系统。ICT Express 6，3（2020年），204–208。[40] Song Yang、Jiamou Liu 和 Kaiqi Zhao。2022年。GETNext：由轨迹流图增强的 Transformer，用于下一兴趣点推荐。载于第45届ACM SIGIR信息检索研究与发展国际会议（西班牙马德里）（SIGIR '22）。计算机协会，纽约，纽约州，美国，1144 1153 10 1145 3477495 3531983 –。2022 [41] https://doi.org/。/。Guanhua Ye、Hongzhi Yin 和 Tong Chen。2022年。面向个性化预测分析的异构设备间去中心化协同学习框架。（2022年）。[42] Hongzhi Yin 和 Bin Cui。2016年。社交媒体中的时空推荐。Springer。[43] Hongzhi Yin、Bin Cui、Zi Huang、Weiqing Wang、Xian Wu 和 Xiaofang Zhou。2015年。联合建模用户兴趣与移动模式以实现兴趣点推荐。载于第23届ACM多媒体国际会议论文集。819–822。[44] Hongzhi Yin、Yizhou Sun、Bin Cui、Zhiting Hu 和 Ling Chen。2013年。Lcars：一种位置与内容感知推荐系统。载于第19届ACM SIGKDD知识发现与数据挖掘国际会议论文集。221–229。[45] Hongzhi Yin、Weiqing Wang、Hao Wang、Ling Chen 和 Xiaofang Zhou。2017年。用于兴趣点推荐的空间感知分层协同深度学习。IEEE知识与数据工程汇刊 29，11（2017年），2537–2551。[46] Hongzhi Yin、Xiaofang Zhou、Bin Cui、Hao Wang、Kai Zheng 和 Quoc Viet Hung Nguyen。2016年。适应用户兴趣漂移以实现兴趣点推荐。IEEE知识与数据工程汇刊 28，10（2016年），2566–2581。[47] H. Zhang、Z. Hu、W. Qin、M. Xu 和 M. Wang。2021年。用于图像识别的对抗协同蒸馏学习。模式识别 111（2021年），107659。[48] Bolong Zheng、Kai Zheng、Xiaokui Xiao、Han Su、Hongzhi Yin、Xiaofang Zhou 和 Guohui Li。2016年。道路网络上的关键词感知连续k近邻查询。载于2016年第32届IEEE数据工程国际会议（ICDE）。IEEE，871–882。

收稿日期：2007年2月20日；修订日期：2009年3月12日；接受日期：2009年6月5日