基于强化学习的稀疏群智感知参与者招募策略

涂淳钰[1)] 於志勇[1),2)] 韩 磊[3)] 朱伟平[1)] 黄昉菀[1),2)] 郭文忠[1),2)] 王乐业[4),5)]

[1)](福州大学数学与计算机科学学院 福州 350108)

[2)](福建省网络计算与智能信息处理重点实验室(福州大学) 福州 350108)

[3)](西北工业大学计算机学院 西安 710072)

[4)](高可信软件技术教育部重点实验室(北京大学) 北京 100871)

[5)](北京大学计算机学院 北京 100871)

摘 要 稀疏群智感知通过选择部分子区域进行数据采集并推测其他子区域的数据,在保证全局数据质量的同时节省了感知成本.然而现有的稀疏群智感知研究工作总是优先选择具有较高价值的子区域,没有考虑被招募的参与者是否能够采集到所需子区域的数据,也忽略了参与者采集的其它数据价值.为了解决子区域选择具有的局限性,本文从参与者的角度出发,提出了应对稀疏群智感知下的用户招募这一问题的新思路,考虑每个参与者采集到的数据对整个采集任务的贡献程度.鉴于每人每天的移动轨迹基本稳定,而不同人在其各自轨迹上采集的数据具有不同的价值,本文利用这种规律性和差异性,研究如何直接招募可采集到高价值数据的参与者.我们采用强化学习框架解决该问题,将用户招募系统作为强化学习的智能体,并且对招募系统的状态、动作和奖励进行建模.本文中使用深度强化学习算法 Deep Q Network(DQN)来训练回报函数,旨在给出在特定的状态下,判断招募哪些用户是最好的选择.该框架在北京市两个月空气质量和一百多名用户移动轨迹的真实数据集上进行了验证,所提出的用户招募策略相比若干基准策略,在用户数量限定下,可获得更高的数据推测精度.

关键词 群智感知;用户招募;压缩感知;粒子群优化;强化学习

中图法分类号 TP391 **DOI 号** 10.11897/SP.J.1016.2022.01539

Direct Participant Recruitment Strategy in Sparse Mobile Crowdsensing

TU Chun-Yu[1)] YU Zhi-Yong[1),2)] HAN Lei[3)] ZHU Wei-Ping[1)]
HUANG Fang-Wan[1),2)] GUO Wen-Zhong[1),2)] WANG Le-Ye[4),5)]

[1)](*College of Mathematics and Computer Science, Fuzhou University, Fuzhou* 350108)

[2)](*Department of Fujian Key Laboratory of Network Computing and Intelligent Information Processing, Fuzhou University, Fuzhou* 350108)

[3)](*School of Computer Science, Northwestern Polytechnical University, Xi'an* 710072)

[4)](*Key Lab of High Confidence Software Technologies, Peking University, Beijing* 100871)

[5)](*School of Computer Science, Peking University, Beijing* 100871)

Abstract Sparse Mobile Crowdsensing (Sparse MCS) selects a small part of sub-areas for data collection and infers the data of other sub-areas from the collected data. Compared with Mobile Crowdsensing (MCS) that does not use data inference methods, Sparse MCS saves sensing costs while ensuring the quality of global data. However, the existing research works on Sparse MCS only focus on selecting a small part of sub-areas with higher value. It does not consider whether the recruited participants can collect the data of the required sub-areas, and also ignores the value

收稿日期:2021-04-02;在线发布日期:2022-01-18.本课题得到国家自然科学基金(61772136,61972008)、福建省杰出青年科学基金项目(2018J07005)、福建省引导性项目(2020H0008)、福建省大数据分析与处理工程研究中心和北大百度基金资助项目(2019BD005)资助.
涂淳钰,硕士研究生,中国计算机学会(CCF)会员,主要研究方向为群智感知.E-mail:tuchunyu1996@163.com.**於志勇**(通信作者),博士,教授,博士生导师,中国计算机学会(CCF)高级会员,主要研究领域为普适计算和群智感知.E-mail:yuzhiyong@fzu.edu.cn.**韩 磊**,博士研究生,中国计算机学会(CCF)会员,主要研究方向为群智感知.**朱伟平**,博士研究生,中国计算机学会(CCF)会员,主要研究方向为群智感知.**黄昉菀**,博士,讲师,中国计算机学会(CCF)会员,主要研究方向为计算智能、机器学习和大数据分析.**郭文忠**,博士,教授,博士生导师,中国计算机学会(CCF)高级会员,主要研究领域为计算智能及其应用.**王乐业**(通信作者),博士,助理教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为群智感知、隐私保护下的城市数据挖掘和城市迁移学习.E-mail:leyewang@pku.edu.cn.

of other data collected by the participants. In order to solve the limitations of traditional methods in sub-areas selection, this paper starts from the perspective of participants and concentrates on the contribution of the data collected by each participant to the entire collection task. All the data contributions collected by each participant will become the basis for decision-making for the participant's choice. And correspondingly, a new idea to deal with the problem of participant selection under Sparse MCS is proposed. In view of the fact that each person's daily movement trajectory is basically stable, and the data collected by different people on their respective trajectories have different values, this paper uses this regularity and difference to study how to directly recruit participants who can collect high-value data. Furthermore, the participant selection problem considered in this paper is not limited to the data collection in the next cycle, but directly recruits some participants to continue the data collection task in the next multiple cycles. The participant selection problem that spans multiple cycles can be modeled as a dynamic decision-making problem. Since heuristic strategies may fall into a local optimal solution, this paper uses reinforcement learning to solve the participant selection problem: We use the participant selection system as an agent of reinforcement learning, and design the state, action and reward of the reinforcement learning model in detail. Factors such as historical selection participant status, sub-areas data collection status and date are considered in the state. The user number is regarded as an action in reinforcement learning, and the reward is reflected by the final data inference error. In order to avoid the problem of excessive number of actions, this paper sets the action to select only one participant at a time until the maximum number of participants is reached, instead of selecting a group of participants at a time. This paper will discuss in detail the difference between the two action modes. To deal with the explosion of the number of state spaces, we use deep reinforcement learning algorithm Deep Q Network (DQN) to train the Q-function, aiming to give the best choice for judging which participants to recruit in a specific state. This framework was verified on a real data set of air quality in Beijing for two months and the movement trajectories of more than one hundred users. Compared with several baseline policies, our proposed participant recruitment strategy can achieve higher data estimation accuracy under a limited number of users.

Keywords crowdsensing; user recruitment; compressive sensing; particle swarm optimization; reinforcement learning

1 引 言

移动群智感知(Mobile Crowd Sensing,MCS)[1]是一种新兴的时空数据采集模式,它通过招募大量携带移动感知设备的用户执行各种感知任务,例如交通流量监测、空气质量监测等.在常规群智感知中,数据质量通常以覆盖率(即被采集的子区域数/子区域总数)来衡量,为了得到满意的数据质量,往往需要招募大量的用户(即参与者),导致成本居高不下.为了解决这个问题,稀疏群智感知[2]被提出,只需采集价值较高的部分子区域数据,然后以一定精度推测未采集子区域的数据.这种方式能够以较少的感知成本,达到令人满意的以数据精度衡量的数据质量.

稀疏群智感知早期工作主要是选择感知子区域[3-5].数据请求者首先定义一个时空范围和一个可以接受的数据精度下限,感知平台在每一个时隙内,逐个"选择"子区域并发布该子区域对应的感知任务,某些参与者收到任务广播后,顺道或者刻意去执行该任务,提交数据给平台,平台根据部分已经收集到的数据,"推测"其余未被感知的子区域的数据,并判断数据精度是否高于下限,是则进入下一个时隙,否则在本时隙内继续进行"选择"-"推测"的迭代.在这种方式下,子区域选择是由委员会投票[3,6]等主动学习样本选择策略决定的,当把需感知的子区域数量作为代价指标时,的确可以实现降低感知成本的目的.然而,这类工作选择出的子区域是所有子区

域中具有代表性、不确定性或多样性的子集，并没有考虑将由哪几个参与者完成，是顺道还是刻意，那么当感知代价是以参与者数量及其移动距离等指标刻画时，感知代价是不可控的，更不是最优的. 以人为选择对象和代价单元，而不是子区域，好处是：(1)考虑了每个参与者采集到的所有数据对整个采集任务的贡献程度. (2)招募和激励一步到位，不需要再针对选取的每个子区域考虑是否有参与者能够到达该子区域.

因此，我们需要从用户的角度出发，将问题从子区域选择转向用户招募. 现有唯一的稀疏群智感知用户招募工作[7]并没有对招募哪个用户“直接”作出决策，而是将用户招募分三阶段进行：先通过筛选出一批在下一周期覆盖子区域较多的用户；然后通过一些子区域选择方法选出部分有效子区域；利用加权计算，使越大概率经过越多已选择子区域的用户具有越大的权重，最终招募权重较大的几名用户. 然而仅仅通过比较覆盖子区域的多少来筛选用户有可能错失一些重要子区域的信息. 并且，这种用户招募方法也无法避免选择出的有效子区域无法招募到用户的问题. 因此，另一种潜在的方法是不分为三阶段，直接从备选用户中招募有效的任务参与者，当他们顺道经过监测子区域附近时收集该子区域的数据，然后用这些收集到的数据来推测未感知区域的数据.

本文将结合用户每天的日常轨迹对其进行直接招募. 由于人们每天的生活具有一定的规律性，这导致了每人每天的轨迹是基本稳定的. 所以我们招募用户进行数据收集任务并不仅限于下一周期，而是覆盖了之后多个周期. 假设我们招募了两名用户进行数据收集任务，那么之后的多个周期内用户只要经过了监测区域就会自动上传该区域的数据. 在一次任务中，一个用户收集到的数据很有可能是横跨多个周期的不同子区域的数据. 如图 1 所示，图中的每个格子代表一个子区域. 假设每次采集数据的周期为1 h，用户 1 和用户 2 于 8 点到 11 点的时间跨度内在各个子区域间移动. 在 8 点到 8 点 59 分这一周期中，用户 1 和用户 2 开始从各自的轨迹起点出发，并且分别收集到了起点处的感知数据. 随着用户 1 和用户 2 的轨迹变化以及周期的推移，在接下来的两个周期中也采集到了他们各自所经过子区域的感知数据.

由上面的例子可以看出，招募用户的结果直接影响最终在哪些周期收集到哪些子区域数据. 所以，

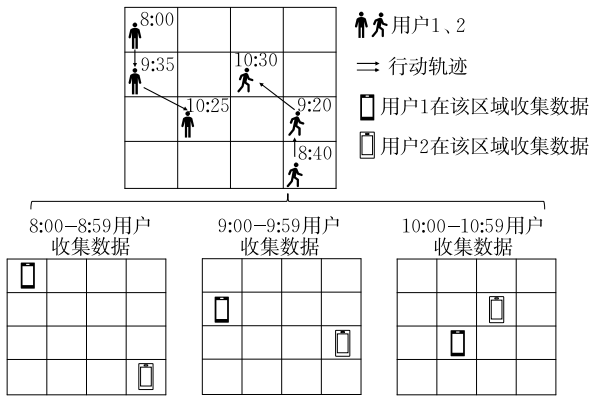


图 1　用户收集数据过程的例子

设计合适的用户招募策略是一个至关重要的问题. 当前招募的用户完成多个周期的数据收集后，用户招募系统将根据数据收集情况以及其他因素决定如何招募下一组用户，因此用户招募问题可以抽象为一个动态决策问题. 近年来，强化学习模型在动态决策问题上取得了显著成效，本文在用户招募策略上也使用了强化学习模型. 强化学习的代理在某种状态下做出决策，通过决策转移到下一状态后获得奖励，经过一系列的决策后获得累计回报. 通过尝试各种不同的决策，获得不同的奖励，代理可以学习到特定状态下的最佳决策.

虽然强化学习是用户直接招募策略的合适模型，我们还需要解决以下几个挑战：(1) 如何用强化学习对本文问题进行具体建模. 状态、动作和奖励的设置将直接影响模型的性能；(2) 如何解决状态和动作空间过大的问题. 将多种复杂的因素考虑进状态中后，状态空间爆炸性增长. 一个动作即招募一组用户，当招募用户的数量增加后，动作的空间也明显增大. 状态空间过大会导致表格法不再适用，虽然可以用神经网络预测累计回报函数来替代表格，但是神经网络的输出节点数量等于动作数量，当动作空间过大神经网络也会面临难以设计和训练的问题. (3) 如何将未来多个周期的情况考虑在内. 传统方法大多仅在当前周期考虑下一周期的情况，而我们的工作将一次招募后用户的聘期从一个周期延长为多个周期. 这些复杂因素使得我们的工作更具挑战性，而不是强化学习的简单应用.

总之，本文工作的贡献如下：

(1) 提出应对稀疏群智感知下的用户招募这一问题的新思路. 区别于现有的“选子区域和选用户分别进行然后简单结合”的三阶段思路，本文“直接”对本次任务需要招募哪些用户进行决策，结合用户的轨迹信息，招募到适合完成跨越多个周期任务的用

户集合.

(2)提出基于强化学习的稀疏群智感知用户招募策略.首先对该问题建立了强化学习的跨越多个周期的用户选取模型,然后使用神经网络和强化学习相结合的 DQN(Deep Q Network)算法进行训练,使智能体学习到不同用户带来的不同分布的时空数据对数据推断算法造成的影响.训练完成后将状态输入神经网络中,可以得到所有对应动作的累计奖励,这种累计奖励不仅考虑了下一步能获取到的即时奖励,还考虑了未来几步可能获得的奖励.在需要作出决策时,强化学习模型往往选择累计奖励最高的动作执行,累计奖励越大代表该用户采集的子区域数据集对最终数据推断的贡献越大.为了进一步提高用户招募的有效性和数据推测的准确性,我们在状态中考虑更多复杂的因素,例如历史招募情况、周期覆盖情况和日期等.

(3)在北京市两个月空气质量和一百多名用户移动轨迹的真实数据集上,验证了我们提出的基于强化学习的稀疏群智感知参与者直接招募策略优于其它基准策略.

本文在第 2 节中讨论相关工作.然后,在第 3 节中提出用户招募问题,并且详细定义用户招募问题的相关概念.在第 4 节中我们详细说明用户招募问题的强化学习模型建立和 DQN 算法的运行过程.第 5 节中我们将评估用户招募策略有效性,最后在第 6 节中做出了总结.

2 相关工作

2.1 用户招募

现有的群智感知工作中,一般需要招募用户覆盖尽可能多的区域和周期.Merkouris 等人[8]将用户招募问题视为最低成本覆盖问题,然后利用贪婪算法解决该问题.Pu 等人[9]提出了一个在线的多重停止问题,来为 MCS 系统动态地选择参与者.在此基础上,Xiao 等人[10]进一步考虑了任务的截止日期和感知时间.刘文彬等人[11]提出了一种基于预测的用户招募策略,关注用户的移动性以有效地选择用户来执行更多的任务.王恩等人[12]还考虑了数据上传的成本,并提出了一种高效的基于预测的用户招募解决方案.

以上的几个工作都集中于尽可能地覆盖更多的感知区域或完成更多的感知任务上,任务分配时需要招募大量用户[11-14],没有涉及利用数据推断降低成本.

2.2 稀疏群智感知

为了进一步地降低成本,一些研究人员提出了稀疏群智感知这一新的群智感知模式[2].

稀疏群智感知将任务分配和数据推断相结合,只需从少量感知子区域收集数据,然后用一些数据推断方法对未感知子区域的数据进行推测.王乐业等人[2,3,6,15]正式提出了稀疏群智感知模式,并且提出了一个具有数据推理、质量评估和子区域选择的框架.韩磊等人[16]利用收集到的稀疏数据中的时空相关性进行建模,对不同时刻的稀疏数据进行了合理的变换和拼接,以保持最新的训练数据.Sun 等人[17]考虑了子区域异质性和恶意参与者问题,提出了一种可信赖且具有成本效益的细胞选择框架,减轻恶意参与者的不利影响.以上工作仅关注子区域选择,没有涉及如何选择具体的参与者来采集这些部分的数据,也没有涉及跨周期的数据采集.

基于王乐业的工作,刘文彬等人[7]提出了包含三个阶段的用户招募策略,通过用户选择和子区域选择筛选出较为有效的用户和子区域,然后利用用户-区域交叉选择出下一周期具有最大权重的用户(越大概率通过已选子区域较多的用户具有较大的权重).然而仅仅通过比较覆盖子区域的多少来筛选用户忽略了该名用户可能收集到的其他数据价值.并且,这种用户招募方法也无法避免地存在选择出的有效子区域无法招募到用户到达的问题.

综上所述,现有的稀疏群智感知研究大多集中于优化子区域选择过程来提高数据推断精确度,却无法解决稀疏群智感知子区域选择问题本身存在的问题:(1)没有考虑被招募的参与者是否能够收集到目标子区域的数据(或是假设了所有区域所有时间的数据均可被收集);(2)未能考虑和充分利用参与者可能收集到的其他数据价值.因此本文将从用户的角度出发,直接决策招募哪些用户在未来多个周期进行感知任务.

2.3 强化学习

跨越多个周期的用户招募问题也可以建模为动态决策问题,而强化学习被认为适于解决动态决策问题.强化学习目前已经被广泛应用于工业制造[18]、仿真模拟[19]、机器人控制[20]、优化与调度[21-22]、游戏博弈[23-24]等领域问题.Mnih 等人[25]提出了第一个深度强化学习模型(DQN),成功地处理了高维状态输入并将其应用于 7 个 Atari 2600 游戏.Silver 等人[26]应用 DQN 并训练了 AlphaGo,这是第一个击败世界级玩家的程序.此外,为了处理部分可观察的状态,Hausknecht

和 Stone[27] 在 DQN 中引入了循环神经网络 RNN，称为 DRQN(Deep Recurrent Q-Learning)，并将其应用于 Atari 2600 游戏. Lample 和 Chaplot[28] 甚至使用 DRQN 来玩 FPS 游戏.

强化学习在群智感知领域中也已经开始被研究人员使用. Xiao 等人[29]将群智感知系统与车辆之间的交互作为一种车辆群智感知游戏，然后他们提出了基于 Q-learning 的策略以帮助群智感知系统做出决策，为车辆分配最优的感知策略. 刘文彬等人[7]的三阶段用户招募策略中，在子区域选择阶段他们使用了强化学习方法为当前周期选择最有效的子区域.

强化学习对于动态规划、组合优化等决策问题的解决已经相当成熟，在稀疏群智感知领域也得到了应用，但只是用于选择子区域，没有用于选参与者. 本文将结合强化学习解决稀疏群智感知中的用户招募问题.

3　系统模型和问题形式化

在第 3 节中，我们首先定义几个关键概念，然后简要介绍用于数据推测的压缩感知算法. 接着我们对稀疏群智感知中的参与者选择问题进行定义. 最后，我们给出一个小规模的例子来更详细地说明该问题.

3.1　定　义

定义 1.　用户集合 U：所有用户的集合用 U 表示，$U=\{u_1,u_2,\cdots,u_i,\cdots,u_n\}$. 其中 u_i 表示用户集中的第 i 个用户，$1\leq i\leq n$.

定义 2.　感知区域集合 A：所有的感知区域用集合 A 表示，$A=\{a_1,a_2,\cdots,a_j,\cdots,a_m\}$. 其中 a_j 表示区域集合中的第 j 个子区域，$1\leq j\leq m$.

定义 3.　招募周期 R' 和任务周期 R：我们每次招募到用户后将有固定数量的感知周期(在本文中提及的“周期”若无特别说明，都为感知周期)对数据进行收集，我们将一次招募并且收集数据所经历的周期集合称为招募周期. 如图 2 所示，招募周期表示为 $R'=\{r_1,r_2,\cdots,r_k,\cdots,r_t\}$，其中 r_k 为该招募周期内的第 k 个周期，$1\leq k\leq t$. 任务周期将由 w 个招募周期组成，本文用 $R=\{R'_1,R'_2,\cdots,R'_v,\cdots,R'_w\}$ 表示任务周期，其中 R'_v 为第 v 个招募周期. 以往的工作常常将招募周期和感知周期设置为同一个周期，即每个感知周期都重新招募一组用户参与任务，这样虽然增加了每个感知周期之间调控的灵活度，但需要大量的招募成本和备选用户数量. 本文认为在获取用户移动规律的前提下，一个招募周期可维持多个感知周期，实现“跨周期”的用户招募可减小用户切换频率，提高群智感知系统可用性. 招募周期过长会减弱时空覆盖的均匀程度从而影响精度，过短则需要招募大量的用户造成高额成本. 考虑到数据集分布规律，在本文的实验中任务周期设置为一个月，招募周期为一天，感知周期为一小时.

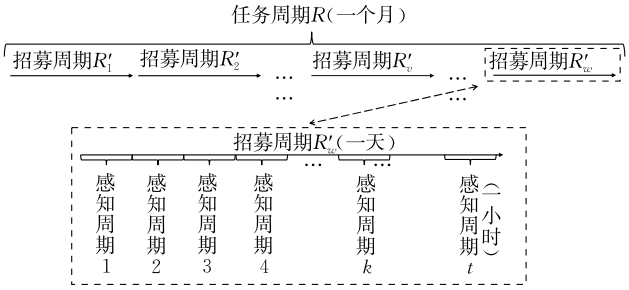


图 2　周期、招募周期和任务周期关系图

定义 4.　用户轨迹：用户 i 的轨迹可以表示为序列 $J_i=\{a_{i1},a_{i2},\cdots,a_{ik},\cdots,a_{it}\}$，序列 J 中有 t 个元素，t 为定义 3 中提到的招募周期拥有的周期数. a_{ik} 为用户 i 在该招募周期的第 k 个周期内所在的子区域，因此 a_{ik} 属于感知区域集合 A. 值得注意的是，用户并不是在所有的周期都经过我们需要感知的子区域. 当用户未经过我们需要感知的子区域时，$a_{ik}=\varnothing$.

定义 5.　收集矩阵 $\boldsymbol{S}_{m\times t}$：本文针对每个招募周期内的子区域数据收集情况定义了感知区域数量为 m，感知周期数量为 t 的矩阵，称为收集矩阵 $\boldsymbol{S}_{m\times t}$. 对于子区域 i，若在本招募周期的感知周期 k 中收集到了该子区域的数据，那么收集矩阵 $\boldsymbol{S}$ 的第 i 行 k 列的元素 $S_{i,k}=1$，否则为 0.

定义 6.　真实环境矩阵 $\boldsymbol{E}_{m\times t}$、感知矩阵 $\boldsymbol{E}'_{m\times t}$ 和推断矩阵 $\bar{\boldsymbol{E}}_{m\times t}$：类似定义 5，真实环境矩阵 $\boldsymbol{E}_{m\times t}$ 中包含本招募周期内所有站点的真实环境数据. 感知矩阵 $\boldsymbol{E}'_{m\times t}$ 中包含用户在本招募周期内收集到的数据，显然 $E'=\boldsymbol{S}\circ\boldsymbol{E}$. 推断矩阵 $\bar{\boldsymbol{E}}_{m\times t}$ 为利用 E' 通过数据推断算法得出的接近于真实环境的矩阵.

3.2　数据推断方法

在稀疏群智感知中，压缩感知在恢复时空数据上具有广泛的应用，它可以从部分收集的传感数据推断真实环境矩阵. 我们借鉴文献[30-31]，也用这个方法. 下面我们将简单介绍压缩感知的原理.

任何矩阵都可以通过奇异值分解为式(1)的形式，我们可以利用 r 个最大的奇异值创建一个秩为 r 的近似矩阵 $\bar{\boldsymbol{E}}$，如式(2)所示.

$$\boldsymbol{E}_{m\times t}=\sum_{i=1}^{\min(m,t)}\sigma_i u_i \boldsymbol{v}_i^{\mathrm{T}} \tag{1}$$

$$\sum_{i=1}^{r}\sigma_i u_i \boldsymbol{v}_i^{\mathrm{T}}=\bar{\boldsymbol{E}} \tag{2}$$

为了求解 r,我们根据低秩性[30]可以解决这个问题:

$$\min rank(\bar{\boldsymbol{E}}) \quad \text{s.t.,}\ \bar{\boldsymbol{E}}\circ\boldsymbol{S}' \tag{3}$$

其中∘表示对应元素相乘的矩阵乘法,$\bar{\boldsymbol{E}}$ 和 $\boldsymbol{E}'$ 分别是数据推断矩阵和实际感知矩阵,r 是收集矩阵.进一步地,由奇异值分解 $\bar{\boldsymbol{E}}=\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}}$,我们可以将上述问题从最小化秩转化为最小化 $\boldsymbol{L}$ 和 $\boldsymbol{R}$ 的 F-范数,式(4)如下所示:

$$\min\lambda(\|\boldsymbol{L}\|_F^2+\|\boldsymbol{R}\|_F^2)+\|\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}}\circ\boldsymbol{S}-\boldsymbol{E}'\|_F^2 \tag{4}$$

为了更好地捕捉感知数据中的时空相关性,我们在优化问题中加入了时间和空间相关性[30,32].

$$\min\lambda_r(\|\boldsymbol{L}\|_F^2+\|\boldsymbol{R}\|_F^2)+\|\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}}\circ\boldsymbol{S}-\boldsymbol{E}'\|_F^2+\lambda_s\|\mathbb{S}(\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}})\|_F^2+\lambda_t\|(\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}})\mathbb{T}\|_F^2 \tag{5}$$

其中 λ_r,λ_s 和 λ_t 是不同相关性的权重,$\mathbb{S}$ 是空间约束矩阵,$\mathbb{T}$ 是时间约束矩阵.$\mathbb{S}$ 矩阵控制同一时刻不同感知区域的数据相关性,$\mathbb{T}$ 矩阵控制同一感知区域不同时刻的相关性.在压缩感知的计算过程中,我们用最小二乘法对 $\boldsymbol{L}$ 和 $\boldsymbol{R}$ 矩阵进行迭代,得到 $\bar{\boldsymbol{E}}=\boldsymbol{L}\boldsymbol{R}^{\mathrm{T}}$.

3.3 问题的形式化

根据第 3.1 节中给出的定义,稀疏群智感知的用户招募问题定义如下.

给定一个具有 m 个子区域、n 个用户和 w 个招募周期的稀疏群智感知任务.在每个招募周期中,我们需要从 n 个用户中招募出 i 个用户.本文用向量 $\boldsymbol{M}^v$ 表示用户是否在招募周期 v 内被招募,若第 k 个用户被招募则 $\boldsymbol{M}^v[k]=1$,否则为 0.用户会在不同的子区域间移动,并且从覆盖的子区域中收集数据.然后我们根据用户收集上来的数据推断未收集区域的数据,最小化数据推断误差:

$$\min\sum_{v=1}^{w}\mathrm{error}(E^v,\bar{E}^v)$$

$$\text{s.t.,}\ |\{k\ |\boldsymbol{M}^v[k]=1,\ 1<k<n,\ 1<v<w\}|=i.$$

下面通过一个例子来详细说明稀疏群智感知的直接用户招募问题.如图 3 所示,考虑我们有 2 个用户和 16 个感知区域,每个招募周期包含 4 个感知周期(r_1,r_2,r_3,r_4),整个任务周期包含 2 个招募周期,每个招募周期仅招募 1 名用户进行数据收集任务.通过用户招募策略,在第一个招募周期我们招募用户 1 在 16 个感知区域中收集数据,用户 1 依次经过子区域 A02、A06、A11 和 A15,在每个周期都收集到了不同子区域的数据;第一个招募周期结束后,通过用户招募策略招募用户 2 进行收集数据,用户 2 也在 16 个感知区域中移动并且收集数据.当任务完成后,我们利用收集到的数据通过数据推断算法推断出未感知的数据.

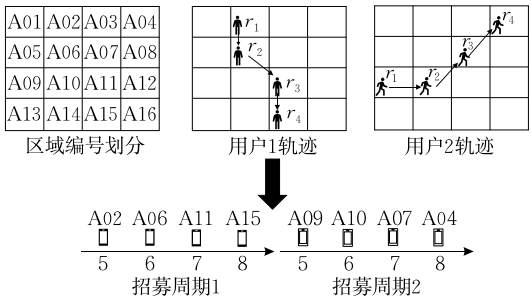


图 3 用户招募问题的数据收集情况

4 方 法

本文提出了直接用户招募框架来解决用户招募问题,如图 4 所示.首先我们针对用户招募问题,结合数据推断算法的特性和数据分布的特点对状态、动作和奖励进行建模,指导用户招募模型有效地学习和处理不同情况下的招募请求.接着,我们使用历史数据训练用户招募模型,训练数据经过数据推断算法处理成强化学习中的奖励反馈给用户招募模型,指导模型对未知情况进行充分探索.最终训练完善的用户招募模型可以招募用户进行收集数据,并且利用数据推断算法推断出接近真实环境数据的全局推断数据.框架中的数据推断模块使用是第 3.2 节中介绍的压缩感知算法.本文在这一部分中将详细说明如何针对用户招募问题对强化学习的各个部分进行建模,以保证用户招募模型能够有效地学习到对数据推断帮助最大的用户招募策略.我们还将详细阐述针对直接用户招募问题的 DQN 算法设置和运行过程.

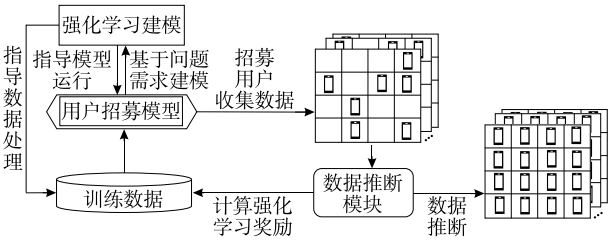


图 4 直接用户招募框架

4.1 状态、动作和奖励建模

群智感知应用场景包括用户招募系统、用户以

及数据请求者三个角色，本文将用户招募系统定义为强化学习的智能体. 用户招募系统在训练过程中通过不断招募不同的用户，学习在各个情况下的最佳用户组合. 明确了智能体后，我们需要对状态、动作和奖励三个关键概念进行建模. 具体而言，在一个包含当前和历史信息的状态下，我们执行了一个动作后会进入下一状态，并且得到一个奖励. 我们可以通过状态、动作和奖励学习状态-动作值函数，它反映的是执行了动作后的累计奖励的期望. 如果一个动作拥有较大的 Q 值，那么它可能是一个较好的选择. 合适的强化学习的建模可以使智能体有效地学习到最佳的策略，如何针对用户招募问题的跨周期和直接招募两个特性，合理地设置状态、动作和奖励是一个至关重要的问题.

(1) 状态包含了多种能影响智能体(用户招募系统)做出下一步决策的因素. 智能体接收状态提供的信息，分析当前情况得知每个动作的好坏程度. 当前和历史招募周期的招募情况直接影响了接下来的招募，具有重要的参考作用. 不同周期数据的覆盖情况影响了数据推断的时间相关性，即使收集到的数据数量很多，但是只是集中在少数几个周期中，那么数据推断将会有很大的误差. 所以我们在状态中加入了当前招募情况对每个感知周期的覆盖情况. 日期也是至关重要的一个因素，有助于强化学习学习数据中的时序模式.

(2) 动作表示了在当前状态下可招募的用户情况的所有可能. 值得注意的是，由于一次从大量用户中选出一组用户的组合数过多(简单考虑 100 个用户挑选 3 个，就有 C_{100}^{3} 种可能)导致了动作空间过多，我们无法很好地对如此庞大的动作空间建模和训练. 因此我们将动作建模为招募一个用户，而不是招募一组用户. 我们需要逐一招募用户，直至招募人数达到上限. 由于动作选择时是逐个添加用户的，如果在一个招募周期内将相同的参与者用不同的顺序来招募，那么最终会导致状态的不同. 但是在同一个招募周期内，用户招募的顺序并不影响最终数据收集的结果. 所以在同一招募周期下，我们每招募一个用户就会对状态进行重新排序.

(3) 奖励表示招募某一用户后对该招募周期的影响程度. 在强化学习中，每次从初始状态到最终状态的过程称为一个回合，本文在设置奖励时将回合是否结束分别考虑，如式(6)所示. 当回合未结束时，智能体获取一个固定常数的奖励，常数的大小根据实际训练时最终的精确度大小调整. 当回合结束时，我们利用数据推断误差来获取奖励. $error$ 的取值范围被设置为[0,100]，当数据推断精度越高时，$error$ 越小，智能体获得的奖励就越大. 注意，本文在训练时假设完整的真实环境矩阵是已知的，在训练时强化学习模型利用真实环境矩阵计算出奖励可以获取当前动作的好坏程度. 而在测试时模型不计算奖励，所以不需要知道完整的真实环境矩阵，这是完全合理的.

$$R=\begin{cases}100-error(E,\bar{E}), & \text{end of Recuitment cycle;}\\ c, & \text{else}\end{cases} \tag{6}$$

4.2　基于 DQN 的用户招募策略

4.2.1　DQN 算法的神经网络设置

本节中将讨论如何针对用户的跨周期和直接招募设计强化学习算法. 在动作设置时我们将动作空间缩小到了一定规模，但是考虑了多种因素的状态导致了状态空间的爆炸性增长. 传统的强化学习算法无法处理过大的状态空间，DQN 结合了神经网络，是一种基于值函数逼近的强化学习算法，该算法可以很好地适用于状态空间过大的强化学习模型. 状态-动作值函数的计算式(7)如下，其中 R 是奖励，它表示在状态 s 下执行动作 a 可以获得的奖励. γ 是计算累计奖励的折扣因子，γ 越接近 1 代表我们更加远视，即更加重视未来的动作可能获得的奖励. 特别地，当 $\gamma=0$ 时 Q 值就仅为执行动作 a 可获得的奖励，此时模型只关注眼前的利益. 通过状态-动作值函数计算出的 Q 值表示了我们在状态 s 下，采取动作 a 所能获得的累计奖励的期望.

$$Q(s,a)=\mathbb{E}\Big[\sum_{k=0}^{\infty}\gamma^{k}R_{t+k+1}\ \big|\ s=S_t,a=A_t\Big] \tag{7}$$

由式(7)我们可以得到状态-动作值函数的贝尔曼方程，可见每个动作的 Q 值不仅由当前动作能获得的奖励决定还要由后续动作能获得的奖励决定.

$$Q(s,a)=\mathbb{E}[R_{t+1}+\gamma Q(S_{t+1},A_{t+1})\,|\,s=S_t,\ a=A_t] \tag{8}$$

DQN 通过拟合一个神经网络来估计 Q 值，该神经网络用来计算每个状态所对应的所有动作的 Q 值. 简单来说，我们将状态输入后，神经网络会输出所有动作对应的 Q 值. 因此，神经网络结构的设置影响了智能体学习状态-动作值函数的有效性. 第 4.1 节的状态设置中提到，状态中包含周期覆盖情况(对应 S_0 在每个周期的覆盖情况)、历史招募用户情况、当前招募用户情况和日期四种因素. 在 DQN 算法中，

用户编号以及日期因素(星期几)用二进制编码表示.每个周期的覆盖情况用 one-hot 编码表示,该周期若有收集到数据则为 1,否则为 0.例如,我们假设考虑了当前和前两个招募周期情况的状态(假设每个招募周期有 6 个周期,备选用户为 15 人)各部分具体情况如图 5 所示,图中还提供了将各状态因素情况转换为状态编码的实例.S_0 部分表示当前招募周期招募的用户情况,当前已经招募了一个用户,还待招募两个用户的部分用全 0 表示.S_{-1} 部分表示上一招募周期招募的 3 名用户编号,S_{-2} 表示再上一个招募周期招募的 3 名用户编号.

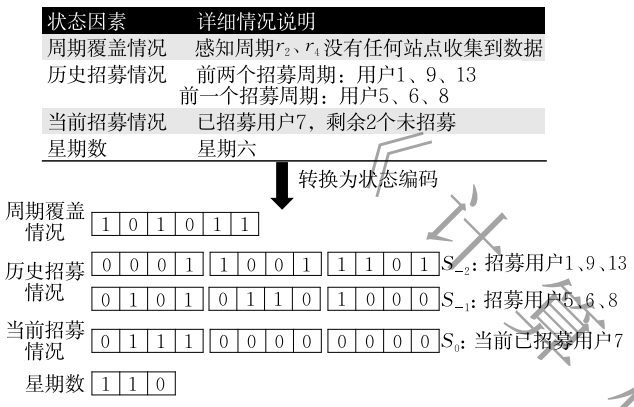


图 5 状态设置例子

如何搭建网络结构是一个重要的问题,已经有大量的研究关注于该问题,但这不是本文关心的主要问题.由于输入的状态由多种因素组成,全连接网络可以处理异构输入,并在我们的状态下捕获综合相关性,所以本文选用具有两个全连接层的网络来作为隐藏层.DQN 中神经网络的输入为当前状态,本文在图 5 中已经详细解释了状态的编码方式.实际实验中的状态节点数量与最大招募用户个数有关,当最大招募用户个数越大时输入层需要的节点数量越多.隐藏层的节点数量通常被设置为输入层节点数量的两倍,激活函数 *relu* 被使用于第二层隐藏层与输出层之间,其余层之间使用激活函数 *tanh*.输出层的每个节点对应着每个动作的 Q 值,因此动作的数量决定着输出层节点的数量,在强化学习模型中,输出层数量等于用户数量.我们使用随机梯度下降方法更新神经网络的参数 θ,损失函数为

$$L(\theta_t)=\mathbb{E}_{\langle S_t,A_t,R_t,S_{t+1}\rangle}[(R+\gamma\max_{A_{t+1}}Q_{\theta_t}(S_{t+1},A_{t+1})-Q_{\theta_t}(S_t,A_t))^2] \tag{9}$$

因此,

$$\nabla_{\theta_t}L(\theta_t)=\mathbb{E}_{\langle S_t,A_t,R_t,S_{t+1}\rangle}[(R+\gamma\max_{A_{t+1}}Q_{\theta_t}(S_{t+1},A_{t+1})-Q_{\theta_t}(S_t,A_t))\nabla_{\theta_t}Q_{\theta_t}(S_t,A_t)] \tag{10}$$

4.2.2 DQN 算法训练过程

基于 DQN 的用户招募策略中的 DQN 模型训练过程如算法 2 所示.具体过程如下:(1)每个任务周期包含固定数量的招募周期,在每个任务周期开始时,算法会随机初始化出一个状态 *state*.*State* 中包含周期覆盖情况、历史招募周期的用户招募情况、当前招募周期的用户招募情况和日期(此时还未招募用户,为全 0 的向量);(2)通过 ε-*greedy* 算法招募一名用户并计算出奖励(见算法 2);(3)在 S_0 中加入本次招募信息,图 6 中展示了 S_0 的变化过程,更新下一状态.并且将当前状态、奖励、下一状态和当前动作加入经验池中;(4)从经验池中随机抽取固定数量的经验,更新神经网络参数,并且将下一状态置为当前状态;(5)判断是否达到了最大招募数量 k,如果已满足 $i=k$,则生成的新状态作为下一招募周期的初始状态,更新周期覆盖情况和日期,如图 7(a).同时如图 7(b)所示,生成新的全 0 向量 $\boldsymbol{S}_1$ 并将之拼接在当前的 S_0 之后,舍弃最远的一次招募周期的用户招募情况;(6)经过固定的步数 C 将 *eval_net* 的参数赋值给 *Target_net*.

算法 1. DQN 模型训练过程.

1. 初始化:经验池容量 N,当前招募数量 $i=0$,最大招募数量 k,对 *eval_net* Q 赋予随机权重 θ,对 *Target_net* Q' 赋予随机权重 $\theta'=\theta$
2. REPEAT(FOR each R'):
3. *state*=$[R,S_{-k},\cdots,S_{-2},S_{-1},S_0,T]$
4. REPEAT(FOR each R):
5. 通过 ε-*greedy* 算法招募一名用户
6. i++,计算 *reward*
7. 更新 S_0,令 *next_state*=$[R,S_{-k},\cdots,S_{-2},S_{-1},S_0,T]$
8. (*state*,*action*,*reward*,*next_state*)存入经验池
9. 从经验池中随机抽取 *minibatch*
10. 通过式(9)、(10).训练 Q 网络的参数 Θ
11. *state*=*next_state*
12. IF $i=k$:

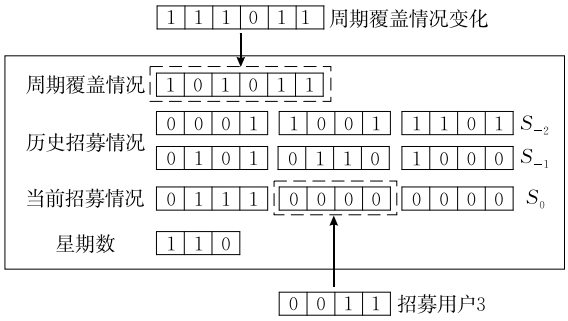


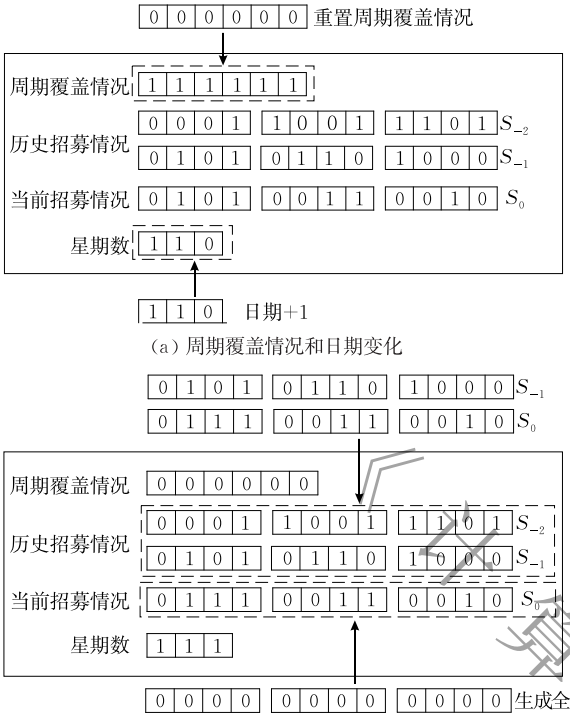
图 6 未满足最大招募数量时的状态变化

13. $S_1=[0,0,\cdots,0]$

14. $state=[R,S_{-k+1},\cdots,S_{-2},S_{-1},S_0,S_1,T]$

15. $i=0$

16. 每 C 步使 $\theta'=\theta$



(a) 周期覆盖情况和日期变化

(b) 历史和当前招募情况变化

图 7　满足最大招募数量时的状态变化

在 DQN 模型训练过程的步 2 中(算法 1 的第 5、6 行)本文使用了 ε-$greedy$ 算法进行决策招募一名用户并且计算奖励,这一如下. 算法 2 的 2～6 行为 ε-$greedy$ 算法,ε-$greedy$ 算法是保证强化学习模型在训练过程中平衡“探索”和“利用”的重要策略. 首先,随机获取一个[0,1]的小数并与预设的 ε 值进行比较. 如果该随机值大于 ε,把当前状态输入 DQN 中的 $eval\_net$,获取该状态下的所有 Q 值,平台将招募一个当前状态下具有最大 Q 值的用户,否则将随机选取一个用户招募. 算法 2 的 7～9 行为计算 $reward$ 的过程. 平台将新招募的用户加入已招募用户集,并从数据集中获取所有已招募用户采集到的感知数据 S. 通过数据推断算法,平台可以获得推断数据 $\bar{\boldsymbol{E}}$. 最终平台将使用式(6),将 $\bar{\boldsymbol{E}}$ 与真实感知数据 E 进行比较计算 reward.

算法 2.　ε-$greedy$ 算法招募用户并计算奖励.

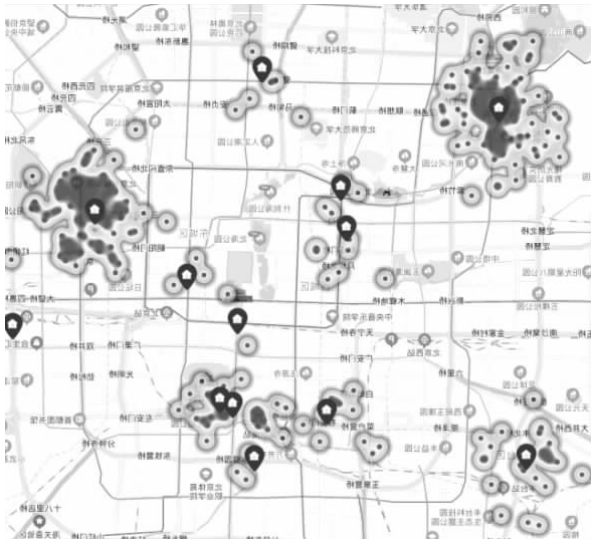
1. 输入:概率值 ε,已招募用户集合 A,数据推断算法 C,当前状态 $state$
2. IF $random()>\varepsilon$:
3. 　$Q=eval\_net(state)$
4. 　$action=argmaxiQ(i)$
5. ELSE:
6. 　$action$ 从所有动作中以均匀分布随机选取
7. $A'=AU\{action\}$
8. 获取用户集 A' 的采集数据 S
9. $\bar{\boldsymbol{E}}=C(S)$
10. 通过式(6)计算 $reward$

5　实　验

5.1　数据集

在实验中本文使用北京市的数据集(包括感知数据集和轨迹数据集)来验证用户招募问题的有效性,如图 8 所示.

(a) 站点位置分布



(b) 四环内用户轨迹覆盖情况热力图

图 8　数据集分布情况图

感知数据集[33]：本文使用了北京市 34 个站点的 AQI、$PM_{2.5}$、CO 和 NO_2 四个数据集进行了大量实验，该数据来自中国环境监测总站的全国城市空气质量实时发布平台，数据的具体情况在表 1 详细给出. 在实验中划分出了以这些站点为圆心，半径为 1000 m 的 34 个圆形感知区域. 站点的位置分布如图 8(a)所示，可以看出，34 个感知区域只在北京市的中心分布比较密集而四周分布比较稀疏.

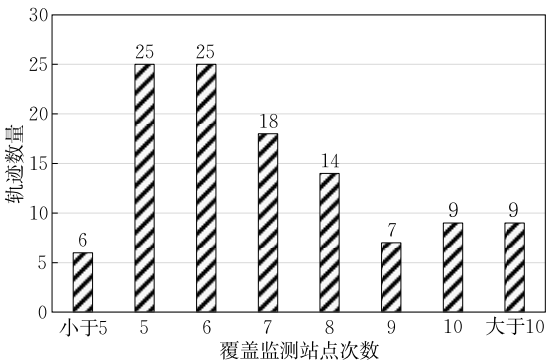
表 1 数据集统计

城市	地理范围	采集间隔	持续时间	均值与标准差
北京	$\pi\times1000\times1000\,m^2$	1 h	2 months	59.91±−38.94 (AQI) 34.72±−34.08 ($PM_{2.5}$) 0.52±−0.37 (CO) 29.06±−18.65 (NO_2)

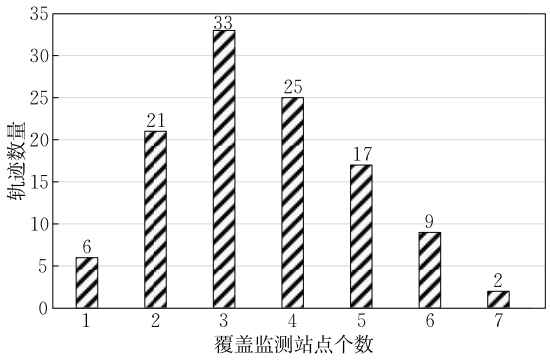
用户数据集[34-37]：Geolife 数据集是由 182 个用户在五年多的时间(2007 年 4 月至 2012 年 8 月)在微软亚洲研究的 Geolife 项目中收集的. 该数据集的 GPS 轨迹由一系列时间戳点表示，每个点包含纬度、经度和高度的信息. 由于实验中使用的数据集是由静态传感器感知的，所以本文假设用户在经过被划分的感知区域时可以通过移动设备成功地感知数据. 基于 Geolife 数据集中的信息，我们剔除了非北京市和不常经过我们设置的感知区域的用户轨迹，最终在感知区域范围内生成了 113 个用户的日常移动轨迹. 每条轨迹表示一个用户一整天(24 h)的行程情况，包含感知周期编号和该周期经过的感知区域两个信息. 四环内的用户对站点的覆盖情况热力图如图 8(b)所示，可以明显看出部分的站点周围用户覆盖程度很高，用户集对站点的覆盖不均匀这一特点正符合用户日常轨迹常集中于住宅和工作地区的分布规律.

每个用户并不是在每个周期内都会经过感知区域，轨迹覆盖监测站点情况的分布如图 7 所示. 由于用户极有可能频繁地在个别站点内活动，重复采集同一站点不同周期的数据，所以我们分别列出了图 9(a)覆盖次数和图 9(b)覆盖站点个数的情况. 可以直观地看出，大部分的用户每天会收集 5～8 次的数据，而经过的监测站点个数显然无法达到该数值，仅有 2 名用户每天会经过 7 个监测站点. 由于各个监测站点只在中心分布得比较密集而四周分布较为稀疏，所以大部分的用户每天仅能覆盖 2～5 个监测站点.

结合每天覆盖监测站点个数，我们可以清楚地发现在用户招募数量较少的情况下，一个招募周期内必然无法获得所有站点任一周期的数据(假设每个人每天覆盖监测站点的个数都为 4，且覆盖的站点不重复，也需要至少 9 个人才能将所有的监测站点覆盖完成). 由于基于矩阵分解的压缩感知算法无法处理矩阵中一行或一列的数据完全丢失的情况，我们在训练和测试时都将先用训练集的均值补全每个站点第一个周期(第一列)的值，然后再使用压缩感知算法进行数据推断.



(a) 轨迹覆盖次数统计图



(b) 轨迹覆盖个数统计图

图 9 轨迹覆盖监测站点情况分布图

本文使用 2020 年 3 月的感知数据对强化学习模型进行训练，在训练时 34 个站点的真实感知数据是已知的，即从 2020-3-1 00:00 到 2020-3-30 23:00，共 30 天. 测试时使用的是 2020 年 4 月的感知数据，从 2020-4-1 00:00 到 2020-4-30 23:00，共 30 天. 在实验中，招募周期 R' 被设置为 1 天，任务周期 R 设置为 30 天，每个周期为 1 h. 我们通过对所有缺失部分的数据计算 MAPE 来评估实验结果，见式(11)，其中 n 为缺失子区域的数量.

$$MAPE=\sum_{i=1}^{m}\sum_{j=1}^{t}\left|\frac{(E[i,j]-\bar{\boldsymbol{E}}[i,j])}{E[i,j]}\right|\times\frac{100}{n} \tag{11}$$

5.2 基准算法和配置

所有的方法可以视为选择候选子区域与确定招募的用户两阶段的结合，所以本文对所有方法以“子区域选择方式-用户招募方式”的形式来命名，如表 2

表 2　基准方法和本文方法名称和简要说明

名称	简要说明
DQN-Greedy	DQN 选点，贪心选人
DQN-Oracle	DQN 选点，选出的点假设均被感知
All-Ran	随机选人
All-PSO	粒子群选人
All-All	选择所有的参与者
All-DQN	我们的方法，DQN 选人

所示. 子区域选择方式或用户招募方式为 All 时表示所有的子区域或用户都被考虑在内. 例如 All-Ran 的选点方式为 All，那么在选子区域阶段中所有的子区域都被考虑在候选集合中（可视为选点阶段无操作），招募用户阶段直接对用户进行随机招募. 而 DQN-Greedy 则表示在子区域选择的时候用 DQN 确定优先度较高的子区域，在用户招募阶段使用贪心算法确定需要招募的用户. 所有方法的具体说明如下：

(1) All-Ran. 在每个招募周期随机对用户进行招募，然后使用用户收集到的数据来对完整的感知数据进行推断.

(2) All-PSO. 元启发式算法是解决组合优化问题的常用方法，其中粒子群优化算法被认为是一种适用性较广的元启发式算法. 我们将使用粒子群算法来对该问题进行求解. 粒子群算法中的一个粒子就是一个解，在用户招募问题中，我们将一个解设置为一组用户的编号，代表需要招募的多组用户. 在粒子群算法得到较优的解（需要招募的用户组合）时，粒子种群中的最优位置就代表了我们需要招募的多组用户，可以直接对这些用户发布数据收集任务. 请注意，本文是直接在测试集上运行 All-PSO 来搜索最优用户组合的. 由于在实际使用时只有用户收集上来的稀疏数据，所以 All-PSO 在实际环境中是无法实现的，本文仅用来做实验对比和分析.

(3) DQN-Greedy. 类似[7]的想法，本文使用 DQN 从每一周期中挑选出 m 个合适的子区域集合，然后从用户集中筛挑选出在招募周期内覆盖子区域次数最多的 k 个用户，最后从 k 个用户中招募覆盖 DQN 筛选的子区域集合最多的一些用户. 在实验中，我们把 m 设置为 3（34 个站点里选 3 个），k 设置为 20.

(4) DQN-Oracle. 与 DQN-greedy 的区别在于仅做子区域选择，然后假设被选择到的子区域的数据都能直接被采集. 由于不考虑用户招募，直接采集有效性最高的部分子区域，所以该方法接近在相同缺失率下的理论上限.

(5) All-All. 招募所有的备选用户，该方法可以视为整个实验的理论上限.

(6) All-DQN. 即本文提出的直接用户招募策略. 在 DQN 中，我们使用了具有两个全连接层的神经网络来估计 Q 值，并用随机权重初始化该网络. 由于强化学习的训练是一个不稳定的过程，过大的学习率不仅会导致神经网络难以收敛，还会导致最终策略的波动较大，过小的学习率将耗费大量的时间进行训练，本文将实验中的学习率设置为 0.001，确保模型可以尽快地学习到较为稳定的决策策略. 同时，为了针对强化学习在训练过程中不稳定这一特点，我们使用了经验回放机制、固定 Q 网络和 ϵ-*greedy* 算法来处理（第 4 节中已详细说明）. 经验回放机制中的经验池容量被设置为 2000，batchsize 为 128. 在训练时把 ϵ 被动态地从 1 调整到 0.1，以确保在刚刚开始训练时，模型可以充分地探索动作空间，当探索到神经网络输出的 Q 值较为稳定在一个范围内时，侧重对较优的动作进行利用确保策略收敛. 由于一次招募一组用户会导致状态空间过大的情况，本文将一个招募周期内招募一组用户分解为了逐次招募，并将之分解为强化学习训练回合中的每个动作. 当折扣因子 γ 越小时，靠后招募的用户在计算 Q 值时的权重越小，为了使每个回合内招募的参与者的权重接近，折扣因子 γ 被设置为了 0.99.

5.3　实验结果

5.3.1　动作设置实验

在第 4.1 节说明动作设置时提到，强化学习模型中的动作表示在下一个招募周期需要招募哪些用户. 然而如果我们将动作设置为一个动作选择一组用户，那么会导致动作数量过多的问题. 为了避免动作数量过多的问题，本文在建模时将动作设计为一个动作就只选择一个用户，即动作数量等于备选用户的总数.

这两种动作方式将直接影响强化学习的训练过程. 对于一次选择一组动作来说，强化学习的动作和状态将在每个招募周期之间转换，执行了动作后状态将跳转到下一招募周期，直到任务周期结束后强化学习的这一训练回合结束. 而对一次选择一个动作来说，强化学习的动作和状态仅在本招募周期之内转换，执行了动作后状态还在本招募周期内，直到本招募周期的用户达到招募上限后回合结束. 简单来说，使用一次选一组用户的动作方式，从第一个招募周期就可以“看”到最后一个招募周期的收益. 使

用一次选一人的动作方式，是在前几个招募周期的数据收集基础上“看”当前招募周期的收益，这种方式利用强化学习使得一次选一个用户的效果近似于一次选一组用户的效果. 所以一个动作是选择一组用户的方法是在招募周期间的强化学习，一个动作是选择一个用户的方法是在招募周期内的强化学习.

为了研究两种动作方式的区别，我们在本实验中使用 4 个站点和 6 个用户的小规模数据集进行使用，招募周期间的强化学习的折扣因子 γ 被设置为 0.9，招募周期内强化学习的折扣因子设置为 0.99. 实验结果如图 10 所示. 可以看到，招募周期内的强化学习（一次一个用户）在训练 150 个回合内就已经达到了较好的性能，而招募周期间的强化学习（一次一组用户）在 1400 个回合附近才首次达到了较优的性能，性能略好于招募周期内的强化学习，但是经过了上千回合的训练后依旧稳定性较差. 这是由于招募周期内的强化学习的一个回合仅在本招募周期内执行，当选人数量达到该招募周期上限后回合就结束，所以较为容易收敛. 而招募周期间的强化学习一个回合跨越了多个招募周期，虽然在训练回合中将每个招募周期间更加紧密地联系在一起，导致性能较好于招募周期内的强化学习. 但是这种方法存在动作数过多和训练代价过大的问题，性能提升也不够显著. 同时，招募周期内的强化学习也没有完全忽视招募周期之间的联系，该方法在状态中考虑了历史招募周期的招募情况，建立了历史招募周期和当前招募周期的联系. 当实验拓展到大规模数据集上时，招募周期间的强化学习建模方法将有动作数量过多、训练代价巨大的缺点. 招募周期内的强化学习可以快速达到较优的性能，是可以接受的一种建模方法.

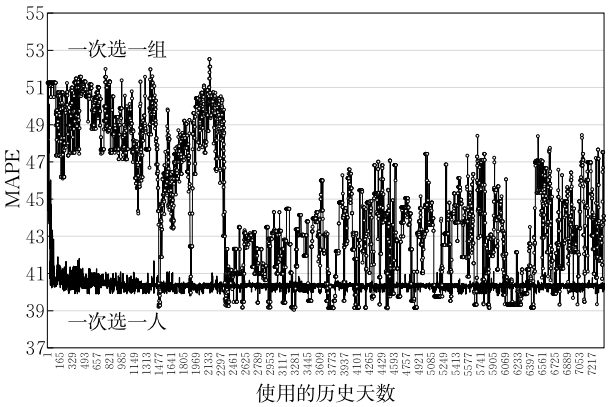


图 10 强化学习训练过程 MAPE 变化图

5.3.2 历史天数影响实验

在第 4 节中提到，状态中考虑了历史招募情况. 由于在招募到用户后数据推断算法需要使用历史收集到的数据和当天收集的数据进行数据推断，本节中将验证多少个历史招募周期的招募情况被考虑在内是适合数据推断算法的.

在数据推断时本文使用了压缩感知算法，历史招募周期数的不同将导致数据推断算法获取的历史信息具有一些差异. 在 AQI、$PM_{2.5}$、NO_2 和 CO 数据集上，我们将使用多个招募周期完整的真实环境矩阵对未来的一个招募周期的感知数据进行推断，然后对推断部分计算 MAPE，实验结果如图 11 所示. 可以看出在没有历史数据帮助推断时，4 个数据集的误差都非常大. 相比于之前，当借助 2 天的数据对下一天进行推断时 MAPE 显著下降，接着继续增加数据只会使得 MAPE 缓慢降低. 这一规律在 AQI、$PM_{2.5}$ 和 NO_2 数据集上表现的较为明显，这 3 个数据集的历史数据达到 2 天后再增加数据对下一天的数据推断影响不大. 而 CO 数据集的 MAPE 随着历史数据的增多下降得较为缓慢，从不使用历史数据帮助推断到使用 2 天的数据帮助推断也使得 MAPE 降低了 15.34，而从 2 到 4 天仅下降了 3.19. 由于历史招募周期的选人情况影响着 RL 状态的数量，随着考虑的历史招募周期越多，状态将呈爆炸性增长. 综合本实验我们认为，在状态中考虑两个历史招募周期可以对下一天的数据推测精度有显著影响，并且状态数量也在可接受范围内，是较为合适的一个选择.

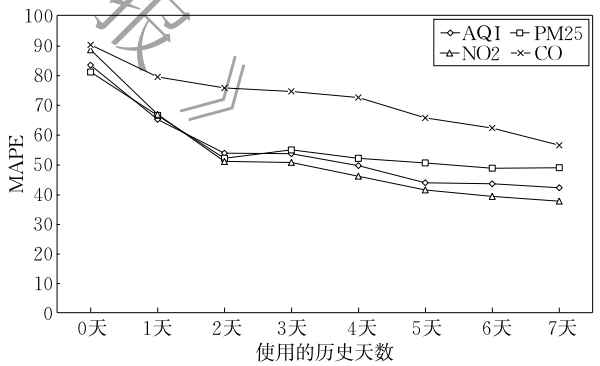


图 11 历史天数对压缩感知的影响

5.3.3 状态有效性实验

本文在强化学习的状态设置中考虑了 4 种因素，包括周期覆盖情况、历史招募周期选人情况、当前选人情况和日期. 为了验证状态中的因素对强化学习模型的影响，我们每次将状态中的一种因素去除，保留剩下的 3 个因素用来训练强化学习模型. 实验结果如图 12 所示，每经过一个完整任务周期（30 天）的数据训练强化学习模型后，我们就使用模型进行测试并记录. 可以看出，历史招募周期的选人情况

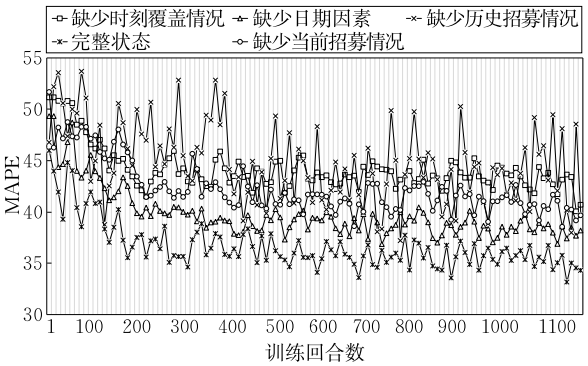
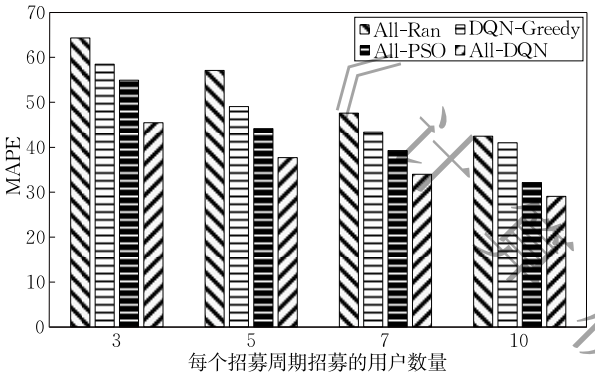


图 12　状态缺失不同因素的训练过程中 MAPE 变化图

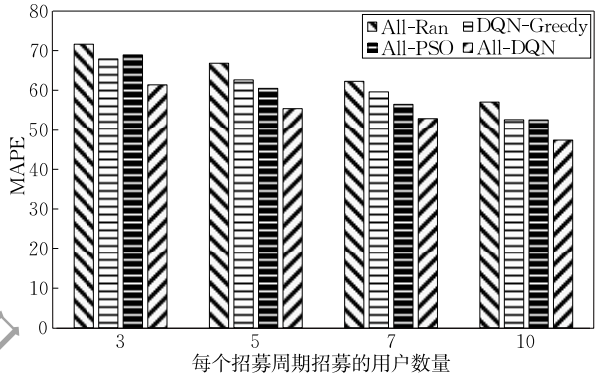
对模型的帮助较为显著，当状态中不加入历史招募周期的选人情况时，模型的测试结果在一个范围内剧烈地波动. 而状态中分别未考虑日期、周期覆盖情况和当前招募情况时能使模型达到一个相对稳定的状态，但是无法做出较好的招募决策，使得利用招募到的参与者收集的数据进行推测后无法得到较优的精度. 当所有的因素考虑在内时，模型可以被训练地较为稳定并且做出较优的决策.

5.3.4　用户招募性能

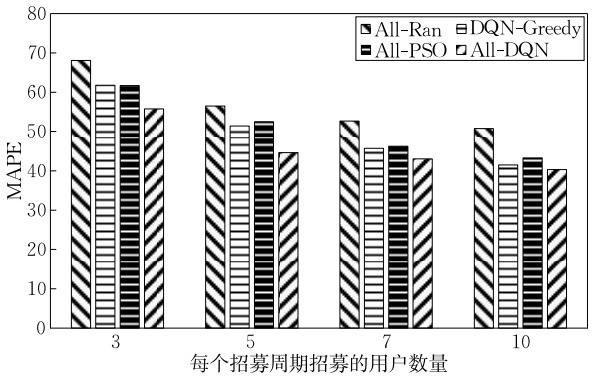
在本节中评估了本文提出的直接用户招募策略在 AQI、$PM_{2.5}$、NO_2 和 CO 四个真实数据集上的性能，我们在每个招募周期分别从 113 个备选用户中挑选 3、5、7、10 个用户来进行感知任务，实验结果如图 13 所示. DQN-Greedy 的性能总是好于 All-Ran，在部分数据集优于 All-PSO，而我们的用户招募策略 All-DQN 总是优于其它三个方法.



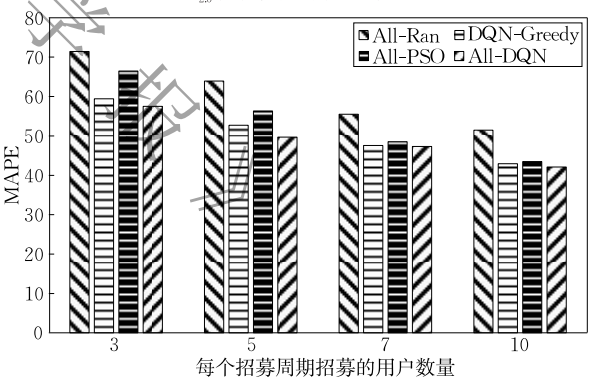
(a) AQI数据集中不同用户数量的MAPE



(b) $PM_{2.5}$数据集中不同用户数量的MAPE



(c) NO_2数据集中不同用户数量的MAPE



(d) CO数据集中不同用户数量的MAPE

图 13　不同数量的招募用户下的 MAPE

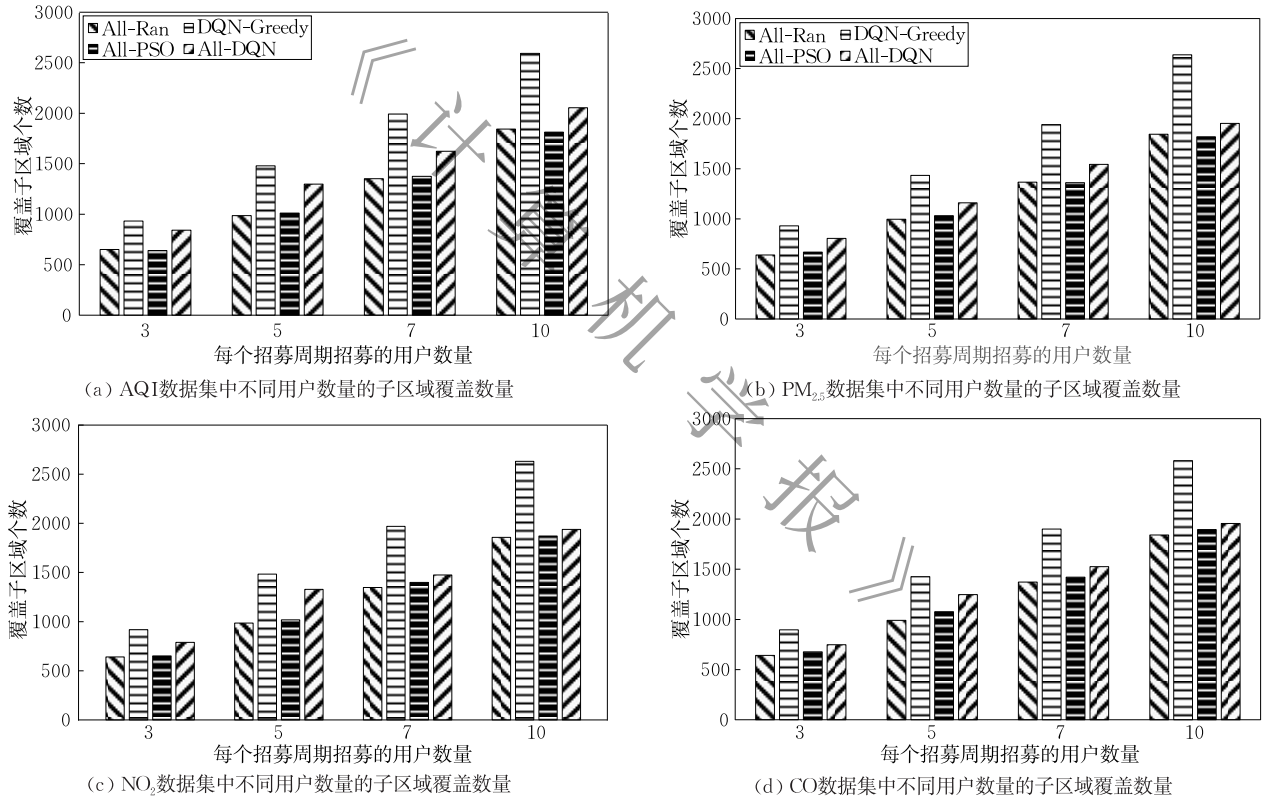
具体来说，对于 AQI 数据集，相比于 All-Ran，All-DQN可以减少29.30%/33.97%/28.57%/31.48%(每个招募周期招募的用户数量为 3/5/7/10)的推理误差. 相比于 DQN-Greedy 和 All-PSO，All-DQN 分别减少了22.25%/23.10%/21.60%/29.06%和17.21%/14.59%/13.35%/9.40%的推断误差. PM2.5 数据集的总体趋势类似 AQI 数据集，All-DQN 比 All-Ran 提升了 14.30%/17.15%/15.25%/16.80%的精确度. 实际上，当每个招募周期招募了 10 个用户来感知时，可以实现较小的推断误差，即 14.11 (AQI)和 7.83($PM_{2.5}$). 标准的 AQI 分级为每 50 个值为一级，$PM_{2.5}$大约 35～40 个值为一级，这样的精度对我们来说是完全可以接受的.

对于另外两个数据集 NO_2 和 CO，DQN-Greedy 的性能比在 AQI 和 $PM_{2.5}$ 数据集上要好，能较优于 All-PSO. 从总体上看，我们提出的用户招募策略 All-DQN 性能仍保持最优. 相比于 All-Ran，我们的用户招募策略减少了 18.13%/20.97%/18.18%/20.50%(每个招募周期招募的用户数量为 3/5/7/10)的推断误差. 相比于 DQN-Greedy 和 All-PSO，分别减

少了 9.76%/13.13%/5.80%/2.80% 和 9.66%/14.97%/6.92%/6.79%.同样地,当我们在每个招募周期招募 10 个用户来感知时,10.99(NO_2)和 0.18(CO)的误差(MAE:29.06+−18.65(NO_2),0.52+−0.37(CO))是完全可以接受的.

5.3.5 覆盖子区域个数对比实验

第 5.3.4 节的实验可以发现,MAPE 随着每个招募周期招募用户数量的增加而下降.这是由于招募用户数量的增加提供了更多有效子区域的信息,从而提高了推断的准确性.但是请注意,并不是单纯的子区域数量增加就会使得 MAPE 降低,我们在记录本次实验数据推测的 MAPE 的同时也记录了本次招募的用户总共覆盖的子区域数量,如图 14 表示.可以看出,由于先从用户集中筛选了部分覆盖子区域最多的用户做备选,所以 DQN-Greedy 在招募用户数量相同的情况下总是能覆盖更多的子区域.然而实验结果表明,随着覆盖子区域的数量增加,数据推断的误差并不一定会降低.这是由于虽然在选择的时候考虑了子区域的有效性,但是备选用户集中的用户并不能将所有的备选子区域覆盖,甚至可能仅仅覆盖了一小部分.并且,仅仅通过覆盖子区域的多少为筛选条件,筛选出的用户集完成任务的质量非常依赖于轨迹集的分布.如果覆盖子区域多的轨迹分散的较为集中在部分区域周围,那么总是会缺失其余区域的数据.而本文提出的方法可以直接学习数据的分布情况,直接对用户招募作出决策,因此不受数据分布的影响,较为通用.



(a) AQI数据集中不同用户数量的子区域覆盖数量

(b) $PM_{2.5}$数据集中不同用户数量的子区域覆盖数量

(c) NO_2数据集中不同用户数量的子区域覆盖数量

(d) CO数据集中不同用户数量的子区域覆盖数量

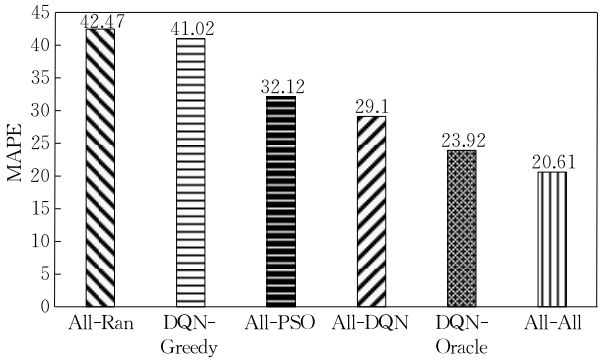
图 14 不同数量的招募用户下的子区域覆盖数量

5.3.6 理想情况对比实验

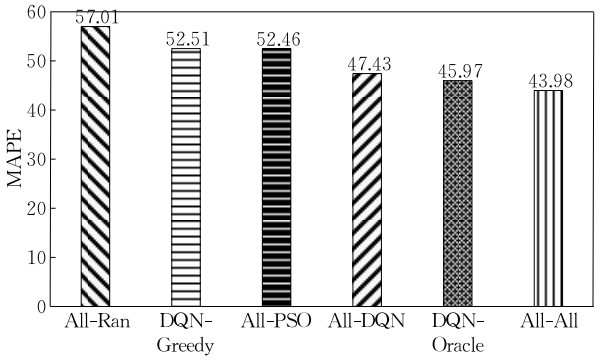
可以注意到当招募用户数量增加时,虽然 All-DQN 减少了推断误差,但下降速度将变得缓慢,甚至接近 All-PSO 和 DQN-Greedy.因此,为了评估我们的用户招募策略的有效性,我们将实验结果与 DQN-Oracle 和 All-All 的实验进行对比.实验结果如图 15 所示,设置每个招募周期招募的用户数量为 10,DQN-Oracle 选出的子区域数量基本与 All-DQN 招募 10 人后收集数据到的子区域数量相同.由于 DQN-Oracle 假设只要被选出的点都能采集,所以可以看作是同等覆盖率下近乎最优的选择.注意,在同等覆盖率下我们的方法无法达到 DQN-Oracle的性能,这是由于一些被选择出需要采集数据的点可能无法被我们的轨迹所覆盖.而 All-All 可以视为整个用户招募策略能够达到的最高上限.

可以看出,在 4 个数据集中 All-DQN 都已经非常接近 DQN-Oracle 的性能.并且在用户招募数量为 10 时,我们的用户招募策略已经快达到模型上限.原因是感知数据集在北京市有非常大的传感面

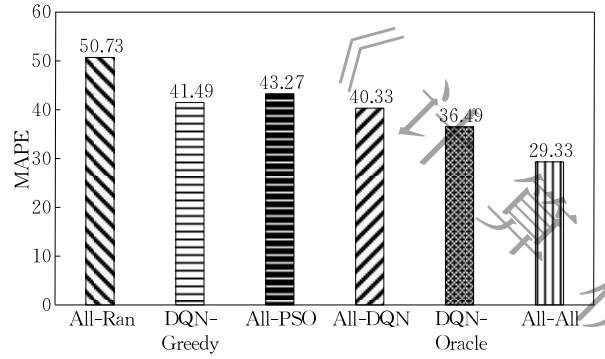
积，分区分布不平衡，城市地区相对密集，郊区非常稀疏. 大部分的用户集中在城市区域内移动，一些偏远区域很难被移动用户覆盖. 并且，在一些周期（例如凌晨 2～4 点）很少有用户在外活动，所以即使增加招募的用户数量，也很难招募到用户覆盖数据较为稀疏的周期. 这也导致 DQN-Oracle 的性能总是能优于 All-DQN，因为它假设所有选出的点都能被感知（不符合实际的理想情况）.



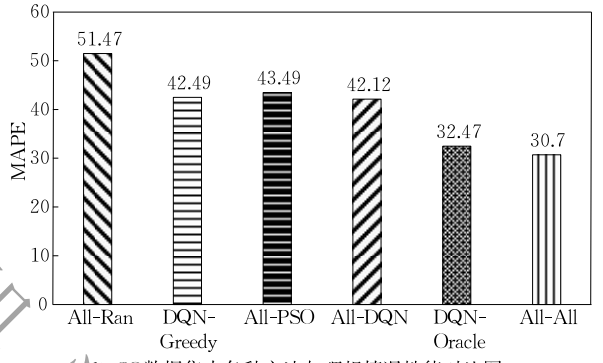
(a) AQI数据集中各种方法与理想情况性能对比图



(b) PM2.5数据集中各种方法与理想情况性能对比



(c) NO_2数据集中各种方法与理想情况性能对比图



(d) CO数据集中各种方法与理想情况性能对比图

图 15　各种方法与理想情况性能对比图

5.3.7　时间分析

表 3 中列出了各个方法在测试时每个招募周期招募 10 名用户的运行时间. 我们的实验平台配备了 Intel(R) Xeon(R) CPU E5-2620 v4@2.10 GHz 和 128 GB RAM. 由于 All-PSO 仅做对比分析使用，当算法迭代结束后才能获得最优用户组合，在实际背景下无法实现，所以未加入时间对比. 对于压缩感知，在测试时上推断一个月的数据大约需要 13.176 s～13.697 s. 如果在训练计算 *reward* 时只需要计算 3 天的数据量，大约只需要 0.385 s. 在确定整个招募周期需要招募的用户时，All-Ran 只需要随机生成用户编号即可，没有任何计算过程导致它的速度是最快的. All-DQN需要0.177s～0.192s，速度远高于需要做子区域选择和用户筛选的 DQN-Greedy. 另外，DQN 模型用 TensorFlow 实现并且可以离线训练，对于具有两个全连接层的神经网络训练大约需要 150 min～200 min.

表 3　运行时间（每个招募周期招募 10 名用户）表格

（单位：s）

	压缩感知	All-Ran	DQN-Greedy	All-DQN
AQI	13.697	0.000891	1.359	0.192
$PM_{2.5}$	13.176	0.000891	1.351	0.191
NO_2	13.361	0.000891	1.307	0.177
CO	13.586	0.000891	1.343	0.178

5.3.8　通用性和局限性分析

稀疏群智感知通过数据推断算法，利用采集到的数据推断完整的全局感知数据，数据推断的准确性直接影响了对任务结果的评价. 由于数据推断的效果与数据集的整体分布模式和采集到的数据位置分布等因素有关，且不同数据推断算法之间也存在着差异性，我们在强化学习训练过程中使用数据推断算法计算的 MAPE 来获得奖励. 这将引导强化学习模型倾向于招募可以采集到对该数据推断算法价值较高数据的用户，以获取较高的奖励. 因此，只要保证获得奖励和最终数据推断的算法一致，本文使用的压缩感知算法完全可以被替换为适合其他类型数据集的数据推断算法，甚至可以将强化学习状态加入更多针对该算法的因素，使用不同的数据推断算法可以训练出针对于该算法的用户招募策略.

本文的用户招募算法在用户招募数量相关方面具有一定的局限性.由于本文将招募一组用户的过程分解为了在一个回合内一次招募一个用户直到人数达到要求,强化学习的折扣因子 γ 在理论上应该设置为 1.但是这会使得强化学习模型在每个状态下都会考虑接下来所有动作能获取的回报(直到回合结束),这将耗费大量的探索训练时间.大部分强化学习模型的 γ 被设置为了 0.9,在 20 个状态-动作的转换后的回报就衰减到了接近 0.1,之后的动作回报几乎会被模型"忽视".本文为了加快训练时间和节约计算资源,将折扣因子 γ 设置为了 0.99,是一种折中的方法.如果具有大量的训练时间以及高性能的硬件设备,可以考虑将折扣因子 γ 设置为 1.否则,在需要招募大量用户时可以适当考虑减小 γ 的值,使强化学习模型忽视离当前状态较远的动作影响.

6 总 结

本文对稀疏群智感知中的用户招募问题进行了研究,在已知用户稳定的通勤轨迹的前提下,一次性为多个连续周期直接招募到高贡献度的用户组合,该用户组合是代价预算内的近似最优组合,因为他们这段时间在小部分位置上采集到的数据,可以利用数据本身的时空相关性,以(相对于其它用户组合)近似最高的准确率恢复出其余未采集数据.我们对用户招募问题的状态、动作和奖励进行建模,提出了针对用户招募的强化学习算法.然后,我们利用强化学习算法直接识别招募哪些用户更加有效,这将导致最终的数据推断更加准确.广泛的实验证明了我们的强化学习模型设置的合理性和用户招募策略的有效性.

在今后的工作中,我们考虑研究用户的移动性预测方法,并且将用户的动向实时地反馈给强化学习模型,以在线的方式完成强化学习的用户招募过程.

参 考 文 献

[1] Guo W, Zhu W, Yu Z, et al. A survey of task allocation: contrastive perspectives from wireless sensor networks and mobile crowdsensing. IEEE Access, 2019, 7: 78406-78420

[2] Wang L, Zhang D, Wang Y, et al. Sparse mobile crowdsensing: Challenges and opportunities. IEEE Communications Magazine, 2016, 54(7): 161-167

[3] Wang L, Zhang D, Pathak A, et al. CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing //Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing. Osaka, Japan, 2015: 683-694

[4] Zhu Z, Chen B, Liu W, et al. A cost-quality beneficial cell selection approach for sparse mobile crowdsensing with diverse sensing costs. IEEE Internet of Things Journal, 2020, 8(5): 3831-3850

[5] Liu W, Wang L, Wang E, et al. Reinforcement learning-based cell selection in sparse mobile crowdsensing. Computer Networks, 2019, 161: 102-114

[6] Wang L, Zhang D, Yang D, et al. SPACE-TA: Cost-effective task allocation exploiting intradata and interdata correlations in sparse crowdsensing. ACM Transactions on Intelligent Systems and Technology (TIST), 2017, 9(2): 1-28

[7] Liu W, Yang Y, Wang E, et al. User recruitment for enhancing data inference accuracy in sparse mobile crowdsensing. IEEE Internet of Things Journal, 2019, 7(3): 1802-1814

[8] Karaliopoulos M, Telelis O, Koutsopoulos I. User recruitment for mobile crowdsensing over opportunistic networks// Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM). Hong Kong, China, 2015: 2254-2262

[9] Pu L, Chen X, Xu J, et al. Crowd foraging: A QoS-oriented self-organized mobile crowdsourcing framework over opportunistic networks. IEEE Journal on Selected Areas in Communications, 2017, 35(4): 848-862

[10] Xiao M, Wu J, Huang H, et al. Deadline-sensitive user recruitment for probabilistically collaborative mobile crowdsensing //Proceedings of the 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS). Nara, Japan, 2016: 721-722

[11] Liu W, Yang Y, Wang E, et al. Prediction based user selection in time-sensitive mobile crowdsensing//Proceedings of the 2017 14th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON). San Diego, USA, 2017: 1-9

[12] Wang E, Yang Y, Wu J, et al. An efficient prediction-based user recruitment for mobile crowdsensing. IEEE Transactions on Mobile Computing, 2017, 17(1): 16-28

[13] Wang J, Wang L, Wang Y, et al. Task allocation in mobile crowd sensing: State-of-the-art and future opportunities. IEEE Internet of Things Journal, 2018, 5(5): 3747-3757

[14] Chen H, Guo B, Yu Z, et al. Crowdtracking: Real-time vehicle tracking through mobile crowdsensing. IEEE Internet of Things Journal, 2019, 6(5): 7570-7583

[15] Wang L, Liu W, Zhang D, et al. Cell selection with deep reinforcement learning in sparse mobile crowdsensing// Proceedings of the 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS). Vienna, Austria, 2018: 1543-1546

[16] Han L, Yu Z, Wang L, et al. Keeping cell selection model up-to-date to adapt to time-dependent environment in sparse mobile crowdsensing. IEEE Internet of Things Journal, 2021, 8(18): 13914-13925

[17] Sun P, Wang Z, Wu L, et al. Trustworthy and cost-effective cell selection for sparse mobile crowdsensing systems. IEEE Transactions on Vehicular Technology, 2021, 70(6): 6108-6121

[18] Gao Yang, Zhou Ru-Yi, Wang Hao, et al. Research on average reward reinforcement learning algorithm. Chinese Journal of Computers, 2007, 30(8): 1372-1378(in Chinese)
(高阳，周如益，王皓等. 平均奖赏强化学习算法研究. 计算机学报，2007，30(8)：1372-1378)

[19] Fu Qi-Ming, Liu Quan, Wang Hui, et al. A novel off policy $Q(\lambda)$ algorithm based on linear function approximation. Chinese Journal of Computers, 2014, 37(3): 677-686(in Chinese)
(傅启明，刘全，王辉等. 一种基于线性函数逼近的离策略 $Q(\lambda)$ 算法. 计算机学报，2014，37(3)：677-686)

[20] Kober J, Bagnell J A, Peters J. Reinforcement learning in robotics: A survey. International Journal of Robotics Research, 2013, 32(11): 1238-1274

[21] Wei Ying-Zi, Zhao Ming-Yang. A dynamic scheduling method of job shop based on reinforcement learning. ACTA Automatica Sinica, 2005, 31(5): 765-771(in Chinese)
(魏英姿，赵明扬. 一种基于强化学习的作业车间动态调度方法. 自动化学报，2005，31(5)：765-771)

[22] Ipek E, Mutlu O, Martnez J F, et al. Self-optimizing memory controllers: A reinforcement learning approach. ACM Sigarch Computer Architecture News, 2008, 36(3): 39-50

[23] Tesauro G. TD-Gammon, a self-teaching backgammon program, achieves master-level play. Neural Computation, 1994, 6(2): 215-219

[24] Nanduri V, Das T K. A reinforcement learning algorithm for obtaining the Nash equilibrium of multi-player matrix games. IIE Transactions, 2009, 41(2): 158-167

[25] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning. arXiv preprint arXiv: 1312.5602, 2013

[26] Silver D, Huang A, Maddison C J, et al. Mastering the game of go with deep neural networks and tree search. Nature, 2016, 529(7587): 484

[27] Hausknecht M, Stone P. Deep recurrent Q-learning for partially observable mdps//Proceedings of the 2015 AAAI Fall Symposium Series. Virginia, United States, 2015: 29-37

[28] Lample G, Chaplot D S. Playing FPS games with deep reinforcement learning//Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. California, United States, 2017: 2140-2146

[29] Xiao L, Chen T, Xie C, et al. Mobile crowdsensing games in vehicular networks. IEEE Transactions on Vehicular Technology, 2017, 67(2): 1535-1545

[30] Zhang Y, Roughan M, Willinger W, et al. Spatio-temporal compressive sensing and internet traffic matrices//Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication. Barcelona, Spain, 2009: 267-278

[31] Kong L, Xia M, Liu X Y, et al. Data loss and reconstruction in sensor networks//Proceedings of the 2013 Proceedings IEEE INFOCOM. Turin, Italy, 2013: 1654-1662

[32] He S, Shin K G. Spatio-temporal adaptive pricing for balancing mobility-on-demand networks. ACM Transactions on Intelligent Systems and Technology (TIST), 2019, 10(4): 1-28

[33] https://quotsoft.net/air/

[34] https://www.microsoft.com/en-us/download/details.aspx?id=52367

[35] Zheng Y, Zhang L, Xie X, et al. Mining interesting locations and travel sequences from GPS trajectories//Proceedings of the 18th International Conference on World Wide Web. Madrid, Spain, 2009: 791-800

[36] Zheng Y, Li Q, Chen Y, et al. Understanding mobility based on GPS data//Proceedings of the 10th International Conference on Ubiquitous Computing. Seoul, Korea, 2008: 312-321

[37] Zheng Y, Xie X, Ma W Y. Geolife: A collaborative social networking service among user, location and trajectory. IEEE Data Engineer Bulletin, 2010, 33(2): 32-39

TU Chun-Yu, M. S. candidate. His research interests focus on mobile crowdsensing.

YU Zhi-Yong, Ph. D., professor, Ph. D. supervisor. His research interests focus on ubiquitous computing and mobile crowdsensing.

HAN Lei, Ph. D. candidate. His research interests focus on mobile crowdsensing.

ZHU Wei-Ping, Ph. D. candidate. His research interests focus on mobile crowdsensing.

HUANG Fang-Wan, Ph. D. candidate, lecturer. Her research interests focus on computational intelligence, machine learning and big data analysis.

GUO Wen-Zhong, Ph. D., professor, Ph. D. supervisor. His research interests focus on computational intelligence and its applications.

WANG Le-Ye, Ph. D., assistant professor, Ph. D. supervisor. His research interests focus on mobile crowdsensing, urban data mining under privacy protection and transfer learning.

Background

More and more mobile portable devices equipped with various sensors can easily collect the noise, temperature, air pressure, air quality and other data around the users. This allows us to rely on Mobile Crowd Sensing (MCS) schema to collect data. Sparse MCS is later proposed to handle the paradox of data quality and sensing cost. It only needs to collect data from a small number of sub-areas to infer the data of the remaining sub-areas, which greatly reduces the number of sensing tasks that need to be performed.

However, existing work in Sparse MCS failed to *directly* determine the users (i. e., participants) that need to be recruited, which causes the cost measured by the number of users to not minimize necessarily. In this work, we tend to fill this gap by directly selecting users who are most helpful for data inference. This is an emerging and important topic in the current field of Sparse MCS. We use the number of users as a constraint, the purpose is to obtain the maximum information under the condition of recruiting the same number of users, so that the error of inferring the global data is minimized. In order to further save costs, we put forward the concept of cross-cycle, that is, recruited users to collect multiple cycles of data. In this case, we need to understand each user's trajectory and the value of sub-areas covered by the trajectory. We model the problem with reinforcement learning, and use it to make decisions about which users need to be recruited. The experiment results show that our method can effectively recruit users who are helpful to improve the accuracy of data inference, and can obtain greater accuracy of data inference compared with the baselines.

This work is supported by the National Natural Science Foundation of China under Grant Nos. 61772136 and 61972008, the Natural Science Foundation of Fujian Province under Grant No. 2018J07005, the Fujian Provincial Guiding Project under Grant No. 2020H0008, the Fujian Engineering Research Center of Big Data Analysis and Processing, and the Project 2019BD005 supported by PKU-Baidu Fund.