

基于MogrifierLSTM的POI推荐算法研究

李昊昱*, 徐一波

河北地质大学信息工程学院, 河北 石家庄

收稿日期: 2024年1月25日; 录用日期: 2024年2月22日; 发布日期: 2024年2月29日

摘要

针对兴趣点推荐研究中忽略用户未来规划对下一POI的影响以及用户当前序列上下文信息学习不充分的问题, 提出了一种基于MogrifierLSTM和未来信息的POI推荐算法——MOGREC。首先, 根据用户签到行为的周期性学习用户历史行为偏好; 其次, 使用LSTM的变体MogrifierLSTM来对当前用户签到序列进行上下文学习; 然后, 利用注意力机制设计用户多步骤未来信息提取器; 最后结合用户当前行为序列以及提取的多步骤未来信息来向用户推荐下一个POI。在Foursquare真实的签到数据集上的实验结果表明, MOGREC算法在长序列数据集SIN上的召回率和NDCG值比最优的CFPRec分别提高了16.17%和18.18%, 验证了MOGREC算法的有效性。

关键词

POI推荐, 未来信息, MogrifierLSTM, 多任务, Transformer

Research on POI Recommendation Algorithm Based on MogrifierLSTM

Haoyu Li*, Yibo Xu

College of Information Engineering, Hebei GEO University, Shijiazhuang Hebei

Received: Jan. 25th, 2024; accepted: Feb. 22nd, 2024; published: Feb. 29th, 2024

Abstract

In order to address the issue of neglecting the impact of user future planning on the next POI and insufficient learning of user current sequence context information in point of interest (POI) recommendation research, we propose a POI recommendation algorithm named MOGREC based on

*通讯作者。

MogrifierLSTM and future information. Firstly, we learn user historical behavioral preferences based on the periodicity of user check-in behavior; Secondly, the MogrifierLSTM variant of LSTM is used to learn context information better on the current user check-in sequence; Then, a user multi-step future information extractor is designed using attention mechanism; Finally, based on the user's current behavior sequence and the extracted multi-step future information, the next POI is recommended to the user. The experimental results on the real check-in dataset of Foursquare show that the MOGREC algorithm has a recall rate and NDCG increase of 16.17% and 18.18% compared to the optimal CFPreC on the long sequence dataset SIN, respectively, verifying the effectiveness of the MOGREC algorithm.

Keywords

POI Recommendations, Future Information, MogrifierLSTM, Multi-Task, Transformer

Copyright © 2024 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 简介

基于位置的社交网络(LBSNs)已经成为一个流行的平台,允许用户分享他们的签到、意见、照片和评论给其他人。随着智能手机的广泛普及,LBSNs的用户数量已经增加到数十亿,这也导致丰富的多模态时空数据呈爆炸性增长。这些数据的增长催生了一个新的推荐系统子领域,即兴趣点(Point of Interest, POI)推荐,该领域在用户和商业角度都引起了广泛的关注。POI推荐技术本质上利用用户的历史签到和其他多模态信息(如POI属性和友谊网络),来为用户推荐下一个最适合的POI。

由于数据的大小和形式以及用户的期望差异很大,这给如何运用大量多模态数据带来了新的挑战,促使研究人员设计新的技术来更好地捕捉移动模式和其他特征(如空间、社交、文本)以提高推荐性能。早期的POI推荐主要关注特征工程和传统的基于机器学习的方法。由于矩阵分解方法在其他推荐系统中的成功应用,研究者开始探索其在POI推荐中的应用。随后,贝叶斯个性化排名方法被引入。此外,支持向量机、协同过滤和高斯建模等传统方法也被用于个性化POI推荐。手工特征设计的困难是限制推荐质量的最主要原因,因此,寻找更好的特征工程方法仍是重要课题,近年来,基于深度学习的方法取代了大多数传统技术。卷积神经网络或循环神经网络等深度学习方法在自动特征提取方面提供了许多优势,消除了手工特征设计的困难。不同深度学习模型的使用,如CNN,RNN、长短期记忆、门控循环单元和自注意力机制,极大地提高了POI推荐模型的性能。

尽管基于RNN框架的方法以数据驱动的方式推荐POI,但仍有几个关键问题尚未解决。1) 基于RNN的方法在短期连续访问中受到限制,它们很难对时间线上远处访问之间的隐式连接进行建模;2) 绝大多数现有算法无法捕获对下一个POI推荐有重要影响的POI和类别、POI与位置之间的跨模式知识;3) 绝大多数模型只考虑过去与当前用户行为而忽略用户未来规划对下一个POI推荐的影响。

针对以上问题,本文提出了基于MogrifierLSTM的POI推荐模型(MogrifierLSTM Based Recommendation, MogRec)模型,主要贡献为:1) 根据用户签到序列的周期性,设计未来信息提取器,结合用户过去、当前和未来序列进行下一POI推荐。2) 引入MogrifierLSTM,增强模型对上下文信息的学习能力,并对时间线上的隐式连接进行建模。

2. 相关研究

上下文信息的捕捉在 POI 推荐中起着至关重要的作用。对于 POI 推荐中上下文信息的捕捉主要分为基于 RNN、基于 LSTM、基于 GRU 三种捕捉方式。

2.1. 基于 RNN 的 POI 推荐方法

Liu [1]等提出了时空循环网络(Spatial-Temporal Recurrent Network, ST-RNN)。该模型扩展了 RNN 模型以用于时空效应。它通过添加一个特定时间和距离的转换矩阵来捕捉时间循环效应和地理效应, 并采用线性插值来训练过渡矩阵。Yang [2]等提出了 Flashback 模型, 该模型利用稀疏的用户移动数据, 关注丰富的时空上下文, 并对 RNN 中的隐藏状态进行回溯。此外, 该模型还采用了历史隐藏状态的加权平均来更好地捕捉时空效应。Zhao [3]等人提出了具有幂等注意模型(Power-Law Attention, ASPPA), 用于学习签到序列的潜在结构。但是 RNN 模型不能很好地捕获长时间的上下文信息, 因此在本文中使用了 MogriflerLSTM 来捕获上下文信息。

2.2. 基于 LSTM 的 POI 推荐方法

Li [4]等人提出时间和多层次上下文注意力模型(Temporal and Multi-Level Context Attention, TMCA), 该模型使用基于 LSTM 的编码-解码网络和三种类型的注意机制: 多级上下文注意(微观、宏观)和时间注意。Sun [5]等人提出了长短期偏好模型(Long and Short-Term Preference Modeling, LSTPM)。该模型将所有签到划分为几个轨迹, 并实际开发了三个模块: 长期偏好模块、短期偏好模块和预测模块。其中, 长期偏好模块使用所有的轨迹, 短期偏好模块使用最后一个轨迹, 并联合预测下一个 POI。Zhang [6]等人提出了交互式多任务学习模型(Interactive multi-task learning, iMTL), 它使用了一个双通道编码器和一个特定任务解码器。双通道编码器包括时间感知的活动和空间感知的位置偏好编码器, 旨在捕捉活动和位置偏好之间的顺序相关性。一些模型修改了基本的 LSTM 模型或使用双向变量, Liu [7]等人提出了时空分层注意力网络(Geographical-Temporal Awareness Hierarchical Attention Network, GT-HAN), 该模型使用了三个要素: POIs 的地理影响、POIs 的地理敏感性和 POIs 之间的距离, 以更好地捕获 POI 领域特定的因素。Lian [8]等提出了基于自注意力网络的地理感知顺序推荐模型(Geographical-Aware Sequential Recommender Based on the Self-Attention Network, GeoSAN)。该模型使用了地理感知自注意网络和地理编码器; Guo [9]等人提出自注意力网络与时空模式结合的模型(Self-Attentive Networks Along with Spatial and Temporal Pattern Learning, SANST), 该模型使用双层 Transformer Network [10]来捕获空间模式。本文受 Zhang [11]等提出了多步骤未来偏好推荐模型(Next Point-of-Interest Recommendation with Inferring Multi-Step Future Preferences, CFPRec), 将用户多步骤未来偏好纳入上下文信息用于推荐下一个 POI。

2.3. 基于 GRU 的 POI 推荐方法

相较于 LSTM, GRU 需要学习的参数更少, 但它有额外的门, 可以解决梯度爆炸和梯度消失的问题。Feng [12]等人提出了 DeepMove 模型。它是一种预测人类流动性的模型, 与 POI 推荐非常相似。它有两个模块: (1) Multi-model Recurrent Prediction Framework: 将时空特征和个人特征联合嵌入到一个密集表示中以提取特征, 然后将该特征输入到 GRU 单元中, 在轨迹序列中建模远程和复杂的依赖关系; (2) Historical Attention Module: 捕捉人类流动的多层次周期性。大多数模型都考虑了不同类型的上下文对用户偏好的相同影响, 但是他们的影响并不相同, 为了解决这个问题。Manotumruksa [13]等人提出了上下文注意循环结构(Context Attention Recurrent Architecture, CARA)。它有四个层: 输入层、嵌入层、循环层、输出层。在循环层中, 该模型使用 GRU。为了捕捉不同情景对用户偏好的影响, 该模型使用了两种门控

机制: (1) 上下文注意门(Contextual Attention Gate): 控制普通情景和过度情景对用户动态偏好的影响; (2) 基于时空的门(Time- and Spatial-based Gate): 考虑连续签到之间的时间间隔和地理距离以控制之前 GRU 单元隐藏状态的影响。本文采用 MogriflerLSTM, 在 LSTM 单元之前使用门控将输入向量与隐藏状态进行交互, 增强捕捉上下文信息的能力。

3. 问题定义

下一个 POI 推荐的问题定义为设 $U = \{u_1, u_2, u_3, \dots, u_u\}$ 为一组用户; 设 $\mathcal{L} = \{l_1, l_2, l_3, \dots, l_l\}$ 为一组 POI; 设 $C = \{c_1, c_2, c_3, \dots, c_c\}$ 为 POI 的一组类别; 设 $T = \{t_1, t_2, t_3, \dots, t_{24}, t_{w0}, t_{w1}\}$ 为一组时间信息(其中一天映射为 24 小时 t_{w0} 为工作日, t_{w1} 为非工作日)。签到记录 $r = (u, l, c, g, t)$ 表示用户 u 在 POI 时刻 t 的签到记录, 该 POI 的类别为 c , 其地理位置信息为 g (经纬度)。对每一个用户 $u \in U$, 设 $S^u = \{S_1^u, S_2^u, S_3^u, \dots, S_n^u\}$ 为该用户的签到轨迹。其中的任一轨迹 $S_i^u = \{r_1, r_2, r_3, \dots, r_{|S_i^u|}\}$, $i \neq n$, 表示该用户一天之中按时间顺序的一组连续签到记录。其中将用户的最新轨迹 $S_n^u = \{r_1, r_2, r_3, \dots, r_k\}$ 视为当前连续签到轨迹。而剩余的轨迹视为过去非连续签到轨迹。

下一个 POI 推荐的研究问题为给出用户 u 的过去行为轨迹 S_{past}^u 和当前行为轨迹 S_{cur}^u , 我们的目标是根据提取的用户多步骤未来偏好来预测用户下一个时间戳的签到行为(即 $t = k + 1$)。

4. MogRec 模型

MogRec 由三个部分组成如图 1 所示, (1) 过去偏好编码器, 用以对用户过去长时间行动轨迹进行建模, 捕捉用户非连续轨迹的上下文相关性, 学习用户的长距离轨迹偏好; (2) 当前偏好编码器, 用以对用户近期的顺序签到轨迹进行建模, 学习用户的当前轨迹偏好; (3) 未来偏好提取器, 用以结合用户的长距离轨迹偏好, 提取出用于的未来多步骤偏好。本模型结合用户的过去长距离非连续签到轨迹, 当前顺序签到轨迹, 以及提取的多步骤未来偏好, 更准确的给出下一个 POI 推荐。

4.1. 过去偏好编码器

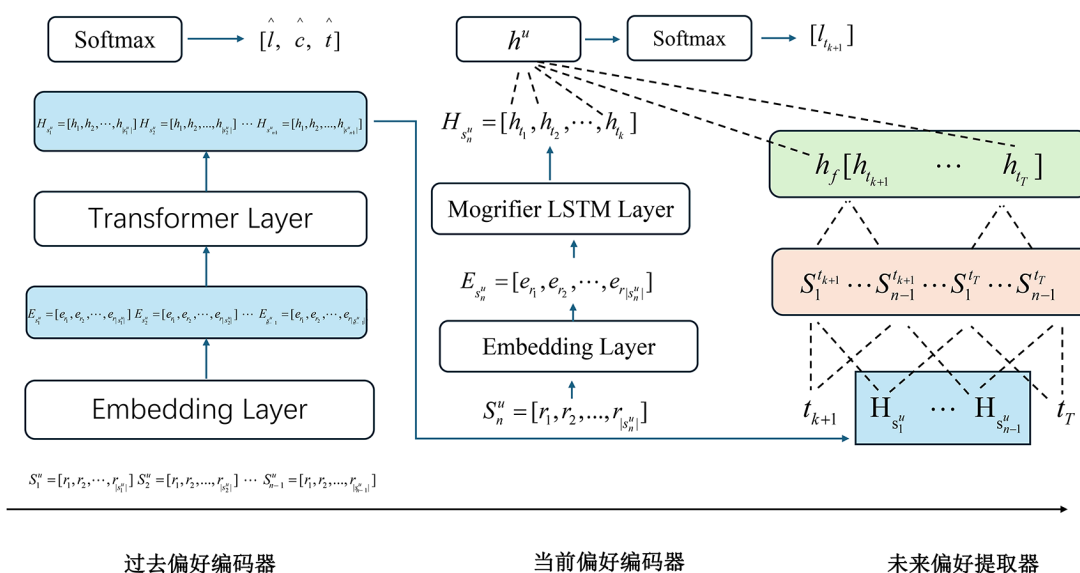


Figure 1. Model framework
图 1. 模型框架

通常情况下, 用户访问 POI 都会出于各种各样的目的, 也就是各种上下文环境, 这也使得用户的签到行为具有复杂性而这种复杂性给学习用户的偏好带来了巨大挑战。幸运的是, 用户的签到行为同样具有周期性, 而这种周期性为用户偏好学习提供了重要的条件。受上下文嵌入方法研究的启发, MOGREC 将 Bi-Transformer 作为过去偏好编码器的首选。(1) Bi-Transformer 作为编码器能够在训练过程中对目标 POI 的左右上下文进行建模, 并可以将 POI 类别、时序、工作日信息和地理信息等上下文都纳入模型中, 能够使得学习的 POI 表示更具表现力。(2) 它能够捕获非连续签到行为之间的上下文关联, 并在一个轨迹内自动聚合具有相关性的签到行为。

因此, 如图 1 中的左边所示, MOGREC 提出一种 Transformer 的拓展, 用以捕获用户的过去偏好, 该编码器能够将用户对 POI 的时空感知动态偏好和关于活动的静态偏好都进行编码。例如, 用户可以在晚餐时间访问具有相同食物(使用相同食物, 即活动)的不同餐厅(即 POI)。

给出用户的过去签到轨迹: $S_i^u = \{r_1, r_2, r_3, \dots, r_{|S_i^u|}\}$, $i \neq n$, $S_i^u \in S_{past}^u$, 每一个签到行为 $r_i = (u, l, c, g, t)$, $r_i \in S_i^u$ 表示为

$$e_{r_i} = u \oplus l \oplus c \oplus t, e_{r_i} \in \mathbb{R}^{5D} \quad (1)$$

其中 \oplus 为联运算; $u, l, c \in \mathbb{R}^D$ 分别为用户, POI 和类别的嵌入表示。 $t \in \mathbb{R}^{2D}$ 表示时间切片为工作日或非工作日, $t = t_k \oplus t_{w0} / t_{w1}$ 。在嵌入层将 S_i^u 转换为 $E_{S_i^u} = [e_{r_1}, e_{r_2}, e_{r_3}, \dots, e_{r_{|S_i^u|}}]$, 然后将其输入到 Transformer 层。

$$\begin{aligned} H_{S_i^u} &= [h_1, h_2, h_3, \dots, h_{|S_i^u|}] \\ &= \text{TransLayer}(E_{S_i^u}) \end{aligned} \quad (2)$$

$$\text{Transformer}(Q, K, V, \Delta^{dist}) = \left(F \left(\frac{QK^T + \Delta^{dist}}{\sqrt{5D}} \right) \right) V \quad (3)$$

其中 $F(\sim)$ 是 Softmax 函数, $\text{Transformer}(\sim)$ 是 Transformer Layer, $H_{S_i^u}$ 为隐藏状态, $h_i \in \mathbb{R}^{5D}$, W_Q, W_K, W_V 分别为查询矩阵, 键值矩阵和权重变换矩阵; $\Delta^{dist} \in \mathbb{R}^{|S_i^u| \times |S_i^u|}$ 为空间关系矩阵。给定从 g_{i_1} 到 g_{i_2} 的距离 $dist(l_i, l_j)$ 。通过 $\Delta_{i,j}^{dist} = (1 + dist(l_i, l_j))^{-1}$ 表示 l_i, l_j 之间的空间影响。

受构建自我监督训练目标以实现掩蔽标记预测的研究启发[14], 设计了三个辅助任务来进一步进行过去非连续签到到轨迹的偏好学习并进行更准确的特征学习。对于任意签到到轨迹 S_i^u , 随机对 20% 的签到记录进行掩蔽, 例如, 对于原轨迹中的签到记录 $r_i = (u, l, c, g, t)$, 使用 $r_m = (u_m, l_m, c_m, g_m, t_m)$ 进行替换, 相对应的嵌入向量和隐藏状态分别用 e_m 和 e_r^{prev} 表示。最后, 我们通过 Softmax 层对隐藏状态 h_m 进行解码, 获得原始位置 l 、类别 c 和时间 t 的多任务预测。

$$\hat{l} = F(h_m W_l), \hat{c} = F(h_m W_c), \hat{t} = F(h_m W_t) \quad (4)$$

其中 $W_l \in \mathbb{R}^{5D \times |L|}$, $W_c \in \mathbb{R}^{5D \times |C|}$, $W_t \in \mathbb{R}^{5D \times |T|}$ 分别为三个预测任务的转换矩阵。其损失函数为:

$$\begin{aligned} L^{aux} &= L_l^{aux} + L_c^{aux} + L_t^{aux} \\ &= \sum_{k \in \mathcal{N}_m} \log(\hat{l}_k) + \log(\hat{c}_k) + \log(\hat{t}_k) \end{aligned} \quad (5)$$

其中 \mathcal{N}_m 为所有掩蔽记录的集合。

4.2. 当前偏好编码器

由于 LSTM 在模拟用户短期顺序行为的广泛使用。因此, MORGEC 使用 LSTM 的变体 MogrifierLSTM [13]来学习用户对当前顺序签到行为的时间感知依赖性, 与 LSTM 不同的是, 在 LSTM 计算之前, 交替的让输入向量 e_r 与隐藏状态 e_r^{prev} 进行交互, 增强上下文建模能力。由于 LSTM 中输入与隐藏状态是相互独立的, 他们只在门中进行交互, 在这之前缺乏交互, 而理论上来说, 当前输入应该和前一个时间戳的掩藏状态相关, 所以 LSTM 会导致上下文信息的丢失。

如图 1 中间部分所示, 给定用户 u 的当前轨迹 S_n^u 的每个签到记录 $r_i \in S_n^u$ 都输入到公式(1)得到嵌入向量 e_{r_i} 。相应的, 将当前轨迹的嵌入向量组 $E_{S_n^u} = [e_{r_1}, e_{r_2}, e_{r_3}, \dots, e_{r_k}]$ 输入到 MogrifierLSTM:

$$\hat{e}_{r_i} = 2\sigma(Q^i h_{r_{i-1}}) \odot e_{r_{i-2}}, \text{ for odd } i \in \{1, \dots, k\} \quad (6)$$

$$\hat{h}_{r_i} = 2\sigma(R^i e_{r_{i-1}}) \odot h_{r_{i-2}}, \text{ for even } i \in \{1, \dots, k\} \quad (7)$$

$$h_{r_i} = LSTM(\hat{e}_{r_i}, \hat{h}_{r_{i-1}}), i \in \{1, 2, 3, \dots, k\} \quad (8)$$

其中 $e_{r_{-1}} = e_{r_0}$, $\hat{h}_{r_0} = h_{r_0}$, r 是一个超参数, 如果 $r = 0$, 那么这就是普通的 LSTM 对应于时间戳 t_i, t_{i-2} 的隐藏状态。 \hat{e}_{r_i} 和 \hat{h}_{r_i} 分别为交互后输入到 LSTM 中的“嵌入向量”和“隐藏状态”。通过这种方式, MogrifierLSTM 最终能够输出用户当前偏好的隐藏状态序列, 即 $H_{S_n^u} = [h_{r_1}, h_{r_2}, h_{r_3}, \dots, h_{r_k}]$ 。

4.3. 未来偏好提取器

如图 1 所示, 用户通常会在行动之前进行规划, 未来的多步骤规划大概率会对用户的后续行动轨迹产生影响, 因此融合未来的行为将有助于更准确的捕获用户偏好, 考虑多步骤未来规划有助于预测用户的下次 POI 访问。如图 2 所示, Bob 在轨迹②中, 18:00 去电影院, 19:00 去购物被视为她的未来行为, 但是在为她推荐 17:00 的餐厅时, 这些未来信息往往不会被关注。但是, 一些基于双向变压器的方法在模型训练中编码了过去和未来的行为, 即为目标 POI 进行左右上下文建模。然而, 由于无法确定用户的未来行为, 极大的限制了模型在训练中捕获用户偏好的能力。受到 CFPreC 的启发, 用户的日常签到行为所具有的周期性为推导未来偏好提取器提供了参考。未来偏好提取器拥有两层注意力聚合机制, 用于推断多步骤未来偏好, 从而在预测用户下一个 POI 时隐式的模拟用户的活动计划。

4.3.1. 序列间注意力机制

由于用户的行为轨迹具有周期性(即用户通常在相同的时间上下文中做出相似的行为), 所以采用注意力机制来进行时间感知, 以识别过去轨迹中与未来时间上下文最相关的行为。用户任一过去轨迹 $S_i^u \in S_{past}^u$, 通过过去偏好编码器被编码为隐藏状态矩阵 $H_{S_i^u} = [h_1, h_2, h_3, \dots, h_{|S_i^u|}]$, 其中每个隐藏状态 h_i 表示用户的时空感知动态 POI 和静态活动偏好。未来时间嵌入 t_f 被用作关注过去轨迹的每个隐藏状态的查询向量, 因此未来时间轨迹嵌入被表示为:

$$s_i^{t_f} = \sum_{i=1}^{|S_i^u|} \alpha_i h_i, t_f \in \{t_{k+1}, t_{k+2}, t_{k+3}, \dots, t_T\} \quad (9)$$

$$\alpha_i = \frac{\exp(t_f^T h_i)}{\sum_{i=1}^{|S_i^u|} \exp(t_f^T h_i)}$$

因此, $s_i^{t_f}$ 能够通过从过去轨迹中提取出关于时间戳 t_f 的最相关行为来当作潜在的多步骤未来行为。

4.3.2. 序列内注意力机制

序列内注意层强调未来时间上下文对轨迹内相关行为的影响。除此之外, 用户日常的顺序行为偏好也在改变, 如图 2 中所示的两条日常轨迹。受 CFPRec 的启发, 引入一个序列间注意层来模拟不同轨迹上的顺序行为演变过程。得到关于 t_f 的多步骤未来轨迹(如 $\{s_1^{t_{k+1}}, s_2^{t_{k+2}}, s_3^{t_{k+3}}, \dots, s_{n-1}^{t_{k+n-1}}\}$)后, 采用动态用户嵌入 u 作为查询向量来关注相关的轨迹偏好:

$$h_{t_f} = \sum_{i=1}^{n-1} \beta_i s_i^{t_f}, t_f \in \{t_{k+1}, t_{k+2}, t_{k+3}, \dots, t_T\}$$

$$\beta_i = \frac{\exp(u^T h_i)}{\sum_{i=1}^{|s^{t_f}|} \exp(u^T h_i)} \quad (10)$$

其中 β_i 为注意力得分; h_{t_f} 为时间戳 t_f 上的未来行为偏好。

通过序列内注意力机制和序列间注意力机制, 未来偏好提取器的搭配了用户多步骤未来偏好。

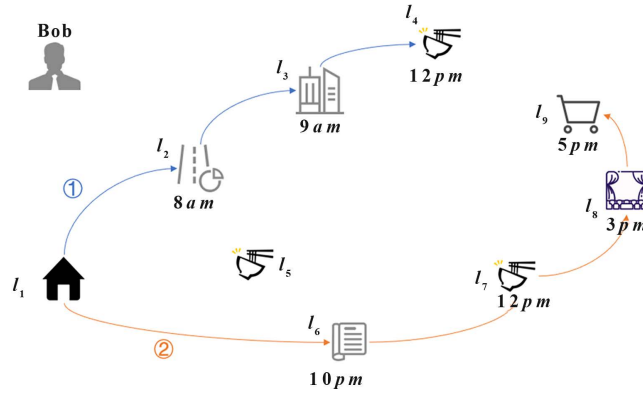


Figure 2. Two of Bob's daily trajectories
图 2. Bob 的两条日常行为轨迹

4.4. 模型训练

通过过去和当前偏好编码器以及未来偏好提取器获得了用户偏好特征 $H_{s_n, f} = [H_{s_n}, H_f]$ 的隐藏状态, 这使得模型能够在训练期间可以对目标 POI 的左右上下文进行建模。

假设用户的下一个 POI 访问不仅受到过去和当前的偏好影响, 还可能受到其多步骤未来偏好的影响。所以用户的最终偏好 (h^u) 通过聚合 $H_{s_n, f}$ 中的每个隐藏状态得到:

$$h^u = \sum_{i=1}^T \omega_i h_i, h_i \in H_{s_n, f}$$

$$\omega_i = \frac{\exp(u^T h_i)}{\sum_{i=1}^T \exp(u^T h_i)} \quad (11)$$

其中 ω_i 为用户动态嵌入 u 和每个隐藏状态 h_i 的注意力得分。随后, 利用学习到的用户偏好 h^u , 使用 Softmax 函数解码得到 $|\mathcal{L}|$ 个 POI 上的概率分布:

$$\hat{y} = F(h^u W_o) \quad (12)$$

其中 $\hat{y} \in \mathbb{R}^{|\mathcal{L}|}$ 为关于 t_{k+1} 的预测概率分布; $W_o \hat{R}^{SD*|\mathcal{L}|}$ 为变换矩阵。所以目标 POI 预测的目标函数为:

$$L^{poi} = -\sum_{i \in \mathcal{N}} \log(\hat{y}_i) \quad (13)$$

其中 \mathcal{N} 为训练样品集合; \hat{y}_i 为对应于第 i 个训练样本的 POI 预测概率的真值。

最后, 给出损失函数:

$$L = L^{poi} + \eta L^{aux} \quad (14)$$

其中 $\eta \in [0,1]$ 为平衡目标 POI 预测和辅助目标的超参数。

4.5. 复杂性分析

在模型训练期间, 过去偏好编码器、当前偏好编码器和未来偏好提取器的计算复杂度分别 $\mathcal{O}((n-1) \cdot |\bar{S}|^2 \cdot D)$, $\mathcal{O}(|\bar{S}| \cdot D^2)$, $\mathcal{O}(D)$ 其中 n 为训练实例的轨迹序列数; $|\bar{S}|$ 为平均轨迹序列长度; D 为嵌入尺寸。由于 $n, |\bar{S}| < D$, 每个训练实例的总体复杂度约为 $\mathcal{O}(|\bar{S}| \cdot D^2)$ 。

5. 实验

5.1. 实验设置

Table 1. Statistics for three datasets

表 1. 三个数据集的统计数据

	用户	POI	签到数	密度
SIN	8648	33,712	355,337	0.12%
NYC	16,387	56,252	511,431	0.06%
PHO	2946	7247	47,980	0.22%

Table 2. Parameter settings for the three datasets

表 2. 三个数据集的参数设置

	D	η	Num_lstm	dropout	mogriifyer_steps
SIN	60	1	3	0.2	5
NYC	40	1	2	0.1	5
PHO	60	1	2	0.2	4

Table 3. Results of SIN

表 3. SIN 数据集结果

	REC@5	REC@10	NDCG@5	NDCG@10	MAP@5	MAP@10
MostPop	0.0107	0.0242	0.0090	0.0126	0.0078	0.0103
BPRMF	0.0302	0.0434	0.0190	0.0318	0.0165	0.026
ST-RNN	0.0824	0.1134	0.0560	0.0665	0.0487	0.0544
ATST-LSTM	0.1114	0.1600	0.0879	0.1238	0.0764	0.1014
MCARNN	0.1382	0.1541	0.1001	0.1334	0.0871	0.1092
iMTL	0.1293	0.1491	0.090	0.1193	0.0783	0.0977
PLSPL	0.1244	0.1423	0.0964	0.1160	0.0838	0.0950
CTLE	0.1754	0.2304	0.112	0.1305	0.0974	0.1069

续表

CFPRec	0.1986	0.2554	0.1360	0.1540	0.1180	0.1252
MogRec	0.2328	0.2967	0.1625	0.1820	0.1414	0.1495
Improve	17.22%	16.17%	19.49%	18.18%	19.83%	19.41%

本文采用 Foursquare 收集的三个数据集, 分别为 Singapore (SIN)、New York City (NYC) 和 Phoenix (PHO) 的 POI 签到数据, 如表 1 所示。本次实验中删除了签到数据少于 10 次的 POI, 并过滤出签到数据少于 5 次的用户数据和签到记录少于 3 次的轨迹数据。然后根据时间戳以 8:1:1 的比例分割每个用户的轨迹序列, 其中 80% 为训练集, 10% 分别为验证集和测试集。最后采用 RECALL@K、NDCG@K 和 MAP@K 评估所有方法的性能。为了进行稳健的比较, 本实验将每个方法运行 10 次后取平均结果作为最终结果。最后通过与其他模型中效果最好的模型进行对比得出提升率(即表 3、表 4、表 5 中的 Improve 项)。

本文将模型与九个基准模型进行比较。(1) MostPop 模型: 通过人气推荐 POI; (2) MPRMF 模型: 基于贝叶斯个性化排名的矩阵分解方法; (3) ST-RNN [1]模型: 基于 RNN 的先驱模型, 结合了下一个 POI 推荐的连续签到之间的空间组合上下文方法; (4) ATST-LSTM 模型, 基于注意力的时空 LSTM 模型, 其关注轨迹中的相关历史签到; (5) MCARNN 模型: 基于多任务上下文感知的 RNN 模型, 其利用空间活动主题进行下一个活动和位置预测; (6) PLSPL [5]模型: 实现了从用户过去和当前行为中捕获用户行为偏好; (7) iMTL [6]模型: 基于多任务的下一个 POI 推荐模型, 其通过双通道编码器对活动和位置的顺序相关性进行建模; (8) CTLE 模型: 基于双向 Transformer 框架的上下位位置嵌入方法; (9) CFPRec [11]模型: 首次基于过去当前行为轨迹和未来轨迹预测的 POI 推荐模型。

本文根据经验找出了所有方法的最佳参数设置。按照 CFPRec 的论文建议设置各种方法的参数并使用 pytorch 实现, 优化器采用 Adam, 学习率设置为 0.0001, 迭代次数为 100, 其他参数如表 2 所示。

Table 4. Results of NY
表 4. NY 数据集结果

	REC@5	REC@10	NDCG@5	NDCG@10	MAP@5	MAP@10
MostPop	0.0121	0.0213	0.0082	0.0123	0.0067	0.0097
BPRMF	0.0334	0.0426	0.0194	0.0289	0.0187	0.0227
ST-RNN	0.1181	0.1662	0.0483	0.1112	0.0662	0.0872
ATST-LSTM	0.1461	0.1849	0.0744	0.1397	0.0819	0.1097
MCARNN	0.1609	0.2182	0.0845	0.1595	0.0902	0.1252
iMTL	0.1576	0.2205	0.0806	0.1588	0.0883	0.1246
PLSPL	0.1526	0.2196	0.0784	0.1557	0.0855	0.1222
CTLE	0.2123	0.2917	0.1234	0.1571	0.1190	0.1233
CFPRec	0.2411	0.3272	0.1599	0.1857	0.1345	0.1449
MogRec	0.2430	0.3283	0.1608	0.1869	0.1363	0.1467
Improve	0.7880%	0.3362%	0.5629%	0.6462%	1.338%	1.242%

Table 5. Results of PHO
表 5. PHO 数据集结果

	REC@5	REC@10	NDCG@5	NDCG@10	MAP@5	MAP@10
MostPop	0.0211	0.0301	0.0144	0.0162	0.0125	0.0134
BPRMF	0.0644	0.0790	0.0324	0.0376	0.0383	0.0312
ST-RNN	0.1641	0.2739	0.1015	0.1521	0.0976	0.1263
ATST-LSTM	0.2090	0.3211	0.1308	0.1714	0.1243	0.1423
MCARNN	0.2522	0.3682	0.1600	0.2002	0.1500	0.1661
iMTL	0.2422	0.3711	0.1647	0.2020	0.1440	0.1677
PLSPL	0.2349	0.3470	0.1627	0.1904	0.1397	0.1580
CTLE	0.3484	0.4870	0.2526	0.2560	0.2072	0.2125
CFPRec	0.4526	0.5632	0.3080	0.3379	0.2740	0.2851
MogRec	0.4529	0.5746	0.3063	0.3420	0.2694	0.2839
Improve	0.0663%	2.024%	-0.5519%	1.213%	-1.679%	-0.4209%

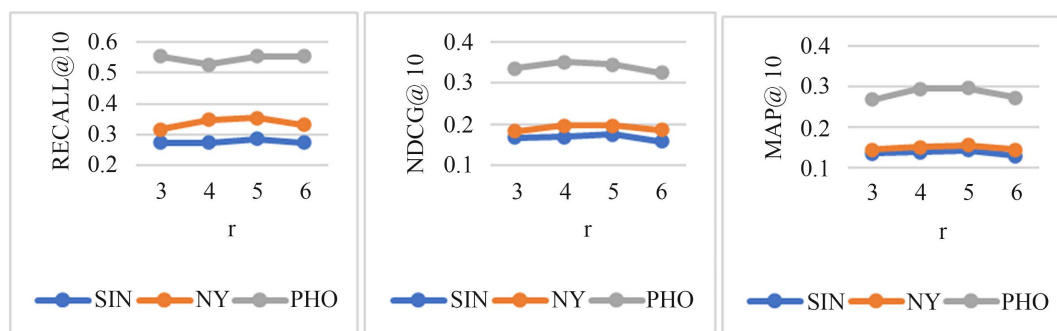


Figure 3. Comparison of R value performance on three datasets

图 3. 三个数据集上 r 取值性能对比

5.2. 结果分析

表 3、表 4、表 5 列出了所有方法的性能。对于三个数据集，MostPop 表现最差，这表明在下一个 POI 推荐中需要进行个性化推荐。BPRMF 模型对比基于 RNN 的模型表相比较差，因为它无法从用户的签到行为中捕获顺序规则。所有基于 RNN 的方法比较如下所述。(1) TST-LSTM 的性能优于 ST-RNN，这表明对顺序行为建模时利用注意力机制可以提高下一个 POI 推荐的质量。(2) MCARNN、iMTL 和 PLSPL 的性能相对较高，这表明将用户的时空感知活动和位置偏好联合建模有助于捕捉用户对其下一个 POI 访问的偏好。(3) PLSPL 的性能较好表明建模过去和当前顺序行为对于更准确的下一个 POI 推荐的重要性，而忽略对未来偏好的建模对比 CFPRec 与 MOGREC 阻碍了其用户偏好学习。(4) CFPRec 与 MOGREC 总体性能上高于其他基线，说明在建模时考虑用户的未来偏好将有利于用户的下一个 POI 推荐。(5) MOGREC 在性能 SIN 与 NYC 数据集上的表现优于 CFPRec，说明在长序列数据集上 MogrifierLSTM 学习用户偏好的能力高于 LSTM。

MOGREC 较之于 CFPRec 的改进主要集中于当前偏好编码器。而 MogrifierLSTM 较于 LSTM 主要是可以将输入与前一个时间戳的隐藏状态在门计算之前进行交互，增强上下们的相关性。所以(1) 相较于 SIN 数据集和 NY 数据集，PHO 的轨迹序列内签到记录较少，所以 MogrifierLSTM 的门交互几乎没有

影响; (2) 对比 SIN 数据集和 NY 数据集, 由于 NY 的签到密度较小, 所以签到记录的提前交互对之影响也较小。

MogrifierLSTM 最重要参数为 `mogrifyier_steps`, 它对应公式(6) (7)中的步骤数。如图 3 所示, 在准备数据时, 我们已经过滤掉了轨迹数量小于 3 的轨迹, 因此我们从 3 开始进行测试。对于长序列数据集 SIN 和 NY, 随着 `mogrifyier_steps` 的增长, `RECALL@10`、`NDCG@10` 和 `MAP@10` 都严格单调递增。对于短序列数据集 PHO, 我们也观察到相同的趋势。这表明 `mogrifyier_steps` 越接近序列长度, 捕捉上下文信心的能力越强。

6. 结语

本文提出了基于 MogrifierLSTM 和未来信息的 POI 推荐算法(MOGREC), 通过用户签到记录的周期性学习到用户的行为偏好, 设计出用户未来偏好提取器, 将未来行为规划纳入考量, 并引入 MogrifierLSTM 来增强上下文的关联性。基于 Foursquare 数据集, 通过比较 MOGREC 算法与其他算法, 发现 MOGREC 算法的召回率和 NDCG 值有所提高。在未来的工作中希望改进未来偏好提取器中的注意力机制, 以期达到更好的推荐效果。

基金项目

国家自然科学基金青年基金(61902105); 河北省科技厅科技支撑计划指令性项目(Z20220032); 河北省教育厅重点项目(ZD2020175)。

参考文献

- [1] Liu, Q., Wu, S., Wang, L. and Tan, T. (2016) Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts. *Proceedings of the AAAI Conference on Artificial Intelligence*, **30**. <https://doi.org/10.1609/aaai.v30i1.9971>
- [2] Yang, D., Fankhauser, B., Rosso, P. and Cudre-Mauroux, P. (2020) Location Prediction over Sparse User Mobility Traces Using RNNs: Flashback in Hidden States! *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, Yokohama, 7-15 January 2021, 2184-2190. <https://doi.org/10.24963/ijcai.2020/302>
- [3] Zhao, K., Zhang, Y., Yin, H., Wang, J., Zheng, K., Zhou, X. and Xing, C. (2020) Discovering Subsequence Patterns for Next POI Recommendation. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, Yokohama, 7-15 January 2021, 3216-3222. <https://doi.org/10.24963/ijcai.2020/445>
- [4] Li, R., Shen, Y. and Zhu, Y. (2018) Next Point-of-Interest Recommendation with Temporal and Multi-Level Context Attention. 2018 *IEEE International Conference on Data Mining (ICDM)*, Singapore, 17-20 November 2018, 1110-1115. <https://doi.org/10.1109/ICDM.2018.00144>
- [5] Sun, K., Qian, T., Chen, T., Liang, Y., Nguyen, Q.V.H. and Yin, H. (2020) Where to Go Next: Modeling Long- and Short-Term User Preferences for Point-of-Interest Recommendation. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 214-221. <https://doi.org/10.1609/aaai.v34i01.5353>
- [6] Zhang, L., Sun, Z., Zhang, J., Lei, Y., Li, C., Wu, Z., Kloeden, H. and Klanner, F. (2020) An Interactive Multi-Task Learning Framework for Next POI Recommendation with Uncertain Check-ins. Yokohama, 7-15 January 2021, 3551-3557. <https://doi.org/10.24963/ijcai.2020/491>
- [7] Liu, T., Liao, J., Wu, Z., Wang, Y. and Wang, J. (2020) Exploiting Geographical-Temporal Awareness Attention for Next Point-of-Interest Recommendation, *Neurocomputing*. <https://doi.org/10.1145/3323873.3325024>
- [8] Lian, D., Wu, Y., Ge, Y., Xie, X. and Chen, E. (2020) Geography-Aware Sequential Location Recommendation. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 23-27 August 2020, 2009-2019.
- [9] Guo, Q. and Qi, J. (2020) Sanst: A Self-Attentive Network for Next Point-of-Interest Recommendation. arXiv: 2001.10379.
- [10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I. (2017) Attention Is All You Need. *Advances in Neural Information Processing Systems*, **2017**, 5998-6008.
- [11] Zhang, L., Sun, Z. and Wu, Z. (2022) Next Point-of-Interest Recommendation with Inferring Multi-Step Future Preferences. *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, Vienna, 23-29 July

2022, 3751-3757. <https://doi.org/10.24963/ijcai.2022/521>

- [12] Feng, J., Li, Y., Zhang, C., Sun, F., Meng, F., Guo, A. and Jin, D. (2018) DeepMove Predicting Human Mobility with Attentional Recurrent Networks. *Proceedings of the 2018 World Wide Web Conference*, Lyon, 23-27 April 2018, 1459-1468. <https://doi.org/10.1145/3178876.3186058>
- [13] Manotumruksa, J., Macdonald, C. and Ounis, I. (2018) A Contextual Attention Recurrent Architecture for Context-Aware Venue Recommendation. *The 41st International ACM SIGIR Conference on Research & Development Information Retrieval*, Ann Arbor, 8-12 July 2018, 555-564. <https://doi.org/10.1145/3209978.3210042>
- [14] Lin, Y., Wan, H., Guo, S. and Lin, Y. (2021) Pre-Training Context and Time Aware Location Embeddings from Spatial-Temporal Trajectories for User Next Location Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, **35**, 4241-4248. <https://doi.org/10.1609/aaai.v35i5.16548>