

武汉理工大学

(申请工程硕士学位论文)

商圈中连续 POI 推荐算法研究 与系统实现

培养单位：计算机科学与技术学院

工程领域：软件工程(两年制)

研究生：黎腾

指导教师：刘永坚 教授

副指导教师：白立华 副编审

2019 年 03 月

商圈中连续 POI 推荐算法研究与系统实现

黎腾

武汉理工大学

分类号_____

密 级_____

UDC _____

学校代码 10497

武汉理工大学

学位论文

题 目 商圈中连续 POI 推荐算法研究与系统实现

英 文 Research and Implementation of successive POI

题 目 Recommendation in Business Circle

研究生姓名 黎腾

指导教师 姓名 刘永坚 职称 教授 学位 博士

单位名称 武汉理工大学 邮编 430070

副指导教师 姓名 白立华 职称 副编审 学位 硕士

单位名称 武汉理工数字传播工程有限公司 邮编 430070

申请学位级别 硕士 学科专业名称 软件工程

论文提交日期 2019 年 3 月 论文答辩日期

学位授予单位 武汉理工大学 学位授予日期

答辩委员会主席 评阅人

2019 年 03 月

独 创 性 声 明

本人声明,所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知,除了文中特别加以标注和致谢的地方外,论文中不包含其他人已经发表或撰写过的研究成果,也不包含为获得武汉理工大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名: _____ 日 期: _____

学位论文使用授权书

本人完全了解武汉理工大学有关保留、使用学位论文的规定,即学校有权保留并向国家有关部门或机构送交论文的复印件和电子版,允许论文被查阅和借阅。本人承诺所提交的学位论文(含电子学位论文)为答辩后经修改的最终定稿学位论文,并授权武汉理工大学可以将本学位论文的全部内容编入有关数据库进行检索,可以采用影印、缩印或其他复制手段保存或汇编本学位论文。同时授权经武汉理工大学认可的国家有关机构或论文数据库使用或收录本学位论文,并向社会公众提供信息服务。

(保密的论文在解密后应遵守此规定)

研究生(签名): _____ 导师(签名): _____ 日 期

摘要

随着定位技术的发展和 O2O 商业模式的火爆,越来越多的研究者将 POI(兴趣点)推荐作为研究课题。不同于传统意义上的推荐系统,POI 推荐具备其自身的特性,对于商圈中 POI 而言,其往往具有很强的区域聚集性。用户在选择签到地点的时候,容易被时间上下文、空间上下文、用户社会关系、POI 主题等所影响,用户在 POI 之间的转移,往往也具有连续性,这些特性决定了 POI 推荐的复杂性。对于商圈中的 POI 推荐,用户通常需要的是实时推荐结果,传统的推荐模型如协同过滤显然无法很好的适应 POI 推荐的场景。针对 POI 的特性,人们提出了连续 POI 推荐的思想,即用户下一个 POI 的签到行为与当前的位置和状态有关。

本文针对商圈 POI 推荐问题,基于用户在 POI 之间的转移特性设计连续 POI 推荐算法,根据不同规模的数据集特点,形成了两种推荐方案。本文的主要研究工作如下:

1) 针对商圈中 POI 的聚集性,提出用 UGSE-LR 模型处理商圈中的 POI 推荐问题,该方法将用户在 POI 之间的转移看作是一个马尔可夫链过程,利用边权个性化 PageRank 计算 POI 的权重,然后利用网格的划分计算 POI 的区域影响,最后引入用户偏好,根据用户的当前位置,为用户生成 POI 的推荐列表。通过实验验证,模型在中等规模的数据集上表现更好。

2) 针对商圈中用户在 POI 之间的转移具有连续性,提出 PEU-RNN 模型来处理商圈中 POI 的连续推荐问题,该方法通过用户的历史签到记录,利用 CBOW 为用户和 POI 创建潜在向量,然后利用 LSTM 模型,预测下一个高概率访问的 POI。实验表明,在大规模数据集上,模型的推荐效果更好。

3) 针对商圈中 POI 的特性,本文根据模型 UGSE-LR,设计并开发了商圈中连续 POI 推荐系统,并引入了标签体系解决了系统的冷启动问题。

关键词: 商圈 POI, 连续推荐, 个性化 PageRank, LSTM

Abstract

With the development of positioning technology and the popularity of O2O business model, more and more researchers have recommended POI (point of interest) as a research topic. Different from the traditional recommendation system, POI recommendation has its own characteristics. For POI in business circle, it often has strong regional aggregation. When selecting check-in location, users are easily influenced by time context, space context, user social relationship, POI topic, etc., and users' transfer between POI is often successive, which determines the complexity of POI recommendation. For POI recommendation in business circle, users usually need real-time recommendation results, and traditional recommendation models such as collaborative filtering obviously cannot adapt to POI recommendation scenarios well. According to the characteristics of POI, the idea of successive POI recommendation is proposed, that is, the check-in behavior of the user's next POI is related to the current location and state.

In this thesis, POI recommendation algorithm is designed based on the transfer characteristics of users between POI in order to solve POI recommendation problem in business circle. According to the characteristics of data sets of different sizes, two kinds of recommendation schemes are formed. The main research work of this paper is as follows:

1) Because of the aggregation of POI in business circles, UGSE-LR model was proposed for POI recommended problems in processing business circle, the method transfers between users in POI as a markov chain process, using the Edge-weighted Personalized PageRank to calculate weight of the POI, and then take advantage of the division of grid computing POI regional influence, finally introduced the user preferences, according to the user's current location, generate the list of recommended POI for users. Experimental results show that the model is more suitable for medium - scale data sets.

2) Because of the successive of users' transfer between POI in business circle, PEU-RNN model is proposed to deal with the successive recommendation of POI in business circle. This method creates potential vectors for users and POI by using

CBOW based on users' historical check-in records, and then predicts the next POI with high probability by using LSTM model. Experiments show that the model performs better in large data sets.

3) According to the characteristics of POI in the business circle, this thesis designs and develops the successive POI recommendation system in the business circle based on the model UGSE-LR, and introduces the label system to solve the cold start problem of the system.

Keywords: POI, successive recommendation, personalized PageRank, LSTM.

目录

摘要	I
Abstract	II
第 1 章 绪论	1
1.1 研究背景及意义	1
1.1.1 O2O 商业模式的巨大市场	1
1.1.2 推荐系统在各行各业中的应用	1
1.1.3 O2O 商业模式中的 POI 推荐	2
1.2 研究现状与存在的问题	3
1.2.1 基本推荐算法分类	3
1.2.2 POI 推荐	7
1.2.3 现有研究存在的问题	11
1.3 研究内容和技术路线	11
1.4 论文的组织结构	12
第 2 章 基于个性化 PageRank 的 POI 连续推荐	14
2.1 问题定义	14
2.2 数据集描述	14
2.3 模型架构	16
2.3.1 用户偏好影响	17
2.3.2 区域影响	17
2.3.3 连续过度影响	19
2.4 POI 之间连续过度影响建模	21
2.4.1 全局 PageRank	21
2.4.2 个性化 PageRank	21
2.5 实验设计与验证	22
2.5.1 实验设计	22
2.5.2 参数的确定	23

2.5.3 网格大小和距离阈值的影响	24
2.5.4 性能比较	25
2.6 本章小结	28
第 3 章 基于神经网络的 POI 连续推荐	29
3.1 模型架构	29
3.2 潜在向量的建立	30
3.3 访问概率预测	32
3.4 POI 推荐	32
3.5 参数测定	33
3.6 实验比较	34
3.7 本章小结	35
第 4 章 系统的设计和实现	36
4.1 需求分析	36
4.1.1 背景	36
4.1.2 可行性研究	36
4.1.3 功能需求	36
4.2 系统设计	38
4.2.1 系统结构设计	38
4.2.2 系统初始化	40
4.2.3 数据库设计	42
4.3 主要界面	44
4.4 本章小结	46
第 5 章 总结与展望	48
5.1 工作总结	48
5.2 展望	48
致谢	50
参考文献	51

第 1 章 绪论

1.1 研究背景及意义

1.1.1 O2O 商业模式的巨大市场

O2O 商业模式在于利用线上资源服务于线下。一般以线下的实体店作为依托，以互联网作为平台。商家通过线上的推广活动，吸引用户到线下进行消费。而对于用户而言，可以通过线上的信息反馈，了解商家的服务质量，资源环境等一系列信息。

数据显示^[1]，2017 年全国电子商务的交易金额为 29.16 万亿元，相比较于去年增长了 11.7%，其中，中国本地生活服务 O2O 的交易金额达 9992 亿元，同比增长 71.5%。在交易额上，口碑和美团点评分列于行业第一第二。

1.1.2 推荐系统在各行各业中的应用

Haubl 和 Trifts 曾经对推荐系统的定义为，一种基于用户的特征学习而进行推荐的软件工具。推荐系统更多的是强调推荐结果的个性化，是信息过滤的一种形式。推荐系统的价值在于它能为用户提供最合适的选择，而不需要用户明确定义他们想要的内容。

亚马逊 (Amazon)^[2] 等公司已经表明，通过将购买和浏览行为的宏观模式进行统计关联，可以大大提高零售体验。亚马逊的成功经常被各种零售商（例如 CD、Barnes 和 Noble）仿效，并且已经有大量关于不同风格的推荐系统在其他各种零售领域产生了很好的推荐效果。Ringo^[3] 根据用户的音乐兴趣推荐音乐项目，淘宝会根据用户的浏览行为推荐相关商品等。

国内字节跳动公司旗下产品“今日头条”客户端，利用数据挖掘算法，以读者兴趣作为导向，颠覆传统的新闻客户端。其最为直观的特点就是，每个用户的客户端都是不一样的，内容更多的是依据用户的历史记录来呈现，这让所有的内容都有出现在首屏的可能。正因为这一点，在 BAT 之后诞生的小巨头，字节跳动拥有强大的竞争力。

基于位置服务的流行也见证了许多生成位置数据和使用位置信息的系统的诞生^[4]。比较简单的如“Shopper’s Eye”^[5]，是一个基于位置的过滤系统，简单地告知用户让人感兴趣的地方，并提供基本信息的介绍。还有“GeoNotes”^[6]这样的系统，允许用户在服务器上发布信息性内容（例如，评论、意见等），以供将来访问该站点的访问者检索。

基于位置的推荐，已经在社交网络中得到了很好的应用。定位技术的发展从根本上增强了社交网络服务^[7]，用户共享他们的位置和与位置相关的内容，海量的位置数据桥接了物理世界和数字世界，对于位置数据的分析和使用可以让我们很好的理解用户的喜好和行为。海量的地理空间数据集也刺激了对寻求促进用户旅行和社会互动的新型推荐系统的研究。比如 Four、Twinkle 和 GeoLife^[8]，在这些系统中，用户可以通过网络轻松地共享他们在现实世界中的位置以及相关内容。例如，经常在同一座山上徒步旅行的人可以被联系起来，通过推荐系统扩展用户的社交网络。地理位置是现实世界与数字世界之间的桥梁，它给传统推荐系统带来了新的机遇和挑战。

1.1.3 O2O 商业模式中的 POI 推荐

O2O 商业模式的出现，让服务提供商，可以很好的获取用户行为数据。Online to Offline 在为用户提供更好的服务的同时，也提供了更多让研究员用以分析的元数据。2013 年，中国的线上销售额仅占零售总额的百分之三，而美国也仅占百分之八^[9]。而 O2O 可以将百分之九十的线下消费，通过互联网的服务，引流到线上的消费行为中。O2O 中的团购模式^[10]，消费者通过登录线上的团购网站，获取商家的折扣信息和其它增值服务，选择线上支付，然后在线下获取商品或者享受服务。这个时候，作为 O2O 的平台，借助于互联网，可以获取到用户的消费行为，通过挖掘用户的偏好信息，为用户生成 POI 的推荐列表，提升用户体验。

作为电商的一个部分，O2O 商业模式有着其自身的独特性。网易云音乐的推荐，其内容为音乐；电商平台淘宝的推荐，其内容为商品。但是对于 O2O 商业模式来说，顾客更多的是在网上完成调研工作或者支付行为，在线下实体店中获取服务。那么，与之相对应的 POI 推荐，也会有着很大的不同。POI 具备其自身的特性，用户在签到的时候，往往会考虑到短时间可达性。然而就 POI 本身而言，不同 POI 之间往往具有连续过度影响，而对于商圈中 POI 而言。其往往具

有很强的区域聚集性，也就是说，同一区域的 POI，往往具有相同的特点，这些特性也决定了 POI 推荐的复杂性。如用户在餐馆消费后，如果按照淘宝或者网易云推荐思路，系统很大可能会继续推荐餐馆之类的商店。这和实际生活会很大的不同。用户在吃完饭之后，更多是会选择休闲娱乐场所进行消费，这里，餐馆到休闲娱乐场所之间的连续过度影响一定是大于餐馆到餐馆的连续过度影响的。对于商圈中的 POI 推荐，就需要结合地理特性和用户签到的上下文，来制定特殊的推荐服务。

1.2 研究现状与存在的问题

1.2.1 基本推荐算法分类

推荐系统按照个性化程度可以划分为两种，一种是非个性化推荐，所有用户获得的推荐项目往往是一样的，不会因为用户的特征不同而有所区别。采用非个性化推荐的情况一般发生在推荐系统的冷启动阶段，对于缺乏用户历史数据，一般数据挖掘算法无效的时候。还有一种是个性化推荐，可以为每个用户定制推荐结果，模型可以用图 1-1 表示：

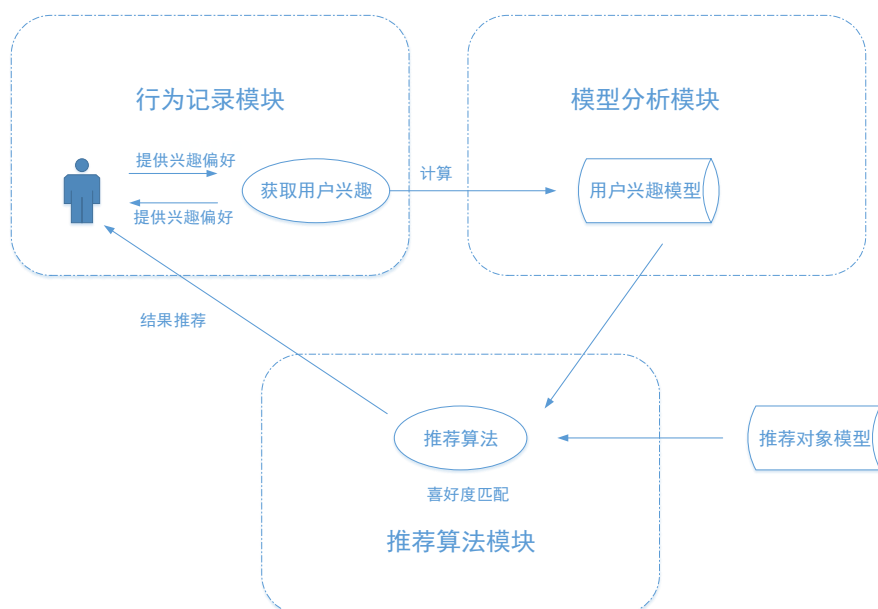


图 1-1 个性化推荐模型图

其中对于行为记录模块，包含了用户的基本信息、行为信息和一些用户的隐式信息。分析模块主要是通过分析用户的历史记录，来构建特征画像。推荐算法

模块主要是通过模型，从项目集合中选择符合用户兴趣的项目集，并最终生成推荐列表。

个性化推荐算法的分类可以用图 1-2 表示：

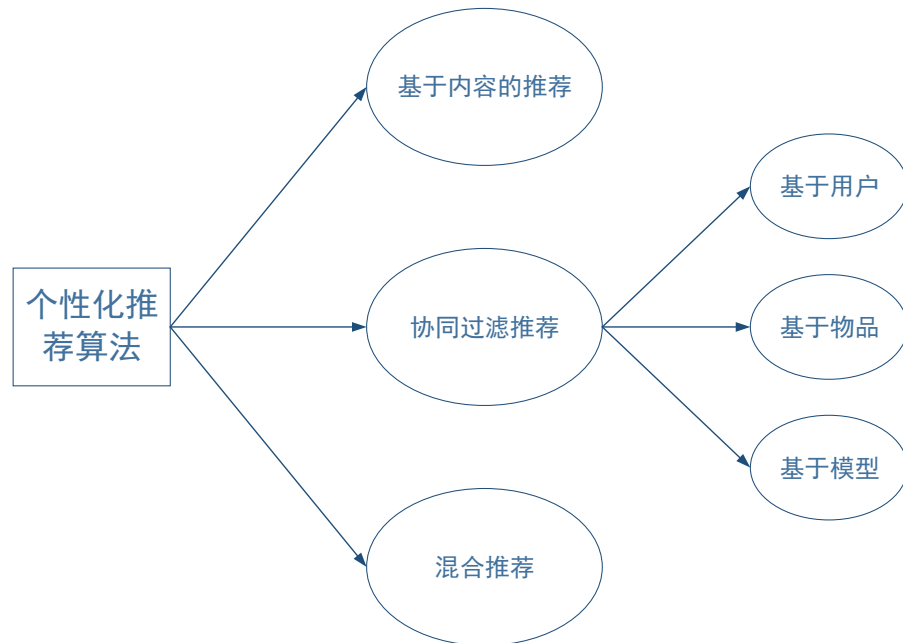


图 1-2 个性化推荐分类

1.2.1.1 基于内容的推荐算法

算法一般依赖于自然语言处理（Natural Language Processing, NLP），方法来源于信息检索。一般都是将项目表示成特征向量，然后根据用户的历史记录计算用户的特征向量，最后通过计算两者的相似程度来生成推荐列表。算法的步骤如图 1-3 所示：



图 1-3 基于内容的推荐算法

1) **Item Representation**: 用关键特征来表示项目属性, 对于项目属性能否被量化, 可以分为结构化属性和非结构化属性。其中, 结构化属性因为可以被量化而可以直接使用, 对于非结构化属性, 一般需要通过 TF-IDF 和 word2vec 等算法进行二次解析。

2) **Profile Learning**: 通过历史数据, 生成用户的特征向量。这里可以简单的用公式表示为:

$$\text{profile} = \frac{\sum_{i=1}^a x_1 + x_2 + \dots + x_a}{a} - \frac{\sum_{i=1}^b y_1 + y_2 + \dots + y_b}{b} \quad (1-1)$$

其中, a 和 b 分别表示喜欢和不喜欢的的项目数量。

而对于实际而言, 我们要解决的是一个有监督的分类问题。可以用机器学习的算法来生成用户模型。

3) **Recommendation Generation**: 通过计算前两步得到的特征向量的相似性, 为用户推荐相关性最大的项目。

1.2.1.2 协同过滤推荐算法

a) 基于内存的协同过滤

算法本身根据处理的思想不同, 又可分为 User-CF (基于用户) 和 Item-CF (基于物品)。其中 User-CF 的思想是: 具有相同偏好的人, 偏爱的东西可能也会相同, 一般用矩阵来记录用户行为, 矩阵项为用户对于物品的打分, 最后比较两个矩阵, 来得出两个用户之间的相似度。但是, 这种思想存在明显的缺点, 特别是当用户量十分庞大的时候, 两个用户之间的相似性计算会花费大量的时间。而对于 Item-CF, 算法的核心在于向用户推荐他之前选择的相似项目, 这个地方是根据评分矩阵, 来计算项目之间的相似度, 他的优点在于基于用户的历史行为, 对于推荐原因有合理的解释。同样的, 在计算项目相似度的过程中, 随着项目种类的增多, 项目之间相似度计算会花费大量时间, 而且无法发掘用户的隐形爱好。

b) 基于模型的协同过滤

我们可以用矩阵分解、张量分解、因子分解机等来实现协同过滤^[11]。

1) 矩阵分解

在推荐系统中, 我们需要将稀疏的用户评分矩阵补全, 根据已有的评分数据, 预测未评分项目的评分, 然后根据估值生成推荐列表。传统的 SVD (奇异值分解) 会对评分矩阵进行补全操作, 一般是全局平均值或者是物品的平均值, 然后

再进行分解降维。但是在实际的操作中，用户数和物品数都是很大的，对一个千万级矩阵进行 SVD 分解，时间会非常的长。而 FunkSVD 采用线性回归的思想，将评分矩阵分解成两个矩阵 P 和 Q ，可以用如下式子表示：

$$M_{m \times n} = P_{m \times k}^T Q_{k \times n} \quad (1-2)$$

BiasSVD 是 FunkSVD 的改版，算法在评分系统中添加了三部分的偏置因素，在某些场景会比 FunkSVD 表现更好。而对于 SVD++，考虑到了用户的隐式信息反馈，是在 BiasSVD 算法上的进一步加强。

贝叶斯个性化排序（Bayesian Personalized Ranking, BPR）也用到了矩阵分解，但是和 FunkSVD 有很大的不同。在实际的推荐场景中，我们要在千万级的商品中推荐个位数的商品给用户，这个时候，需要根据用户偏好，让少数优先级高的商品排名靠前。BRP 是基于矩阵分解的一种排序算法，和 FunkSVD 之类的算法相比，它不是生成全局的分解矩阵，而是针对每个用户，生成商品的喜好排序列表。

2) 张量分解（Tensor Factorization , TF）

与矩阵分解不同，张量分解不单单局限于二维的数据，而是可以更为有效的处理高阶数据，通过补全原始张量，可以得到原始张量中的空缺值，从而生成推荐列表。

对于张量的分解，可以用下面的式子表示：

$$Y = G \times_1 U_1 \times_2 U_2 \times \cdots \times_n U_n \times_N U_N \quad (1-3)$$

其中， U_1, \dots, U_N 是因子矩阵， G 为核心张量，用来决定各个因子矩阵之间交互关系。 \times_n 表示张量和 mode- n 向量的乘积， n 表示张量乘以矩阵的方向。

3) 因子分解机

因子分解机（Factorization Machines, FM）^[12]考虑特征之间的关联，对于稀疏的用户评分矩阵，有很好的处理效果。因子分解机可以用如下式子表示：

$$y(x) = \omega_0 + \sum_{i=1}^n \omega_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \omega_{ij} x_i x_j \quad (1-4)$$

其中， x_i 为输入特征， y 为预测标签， $\omega_0 \in R$ ， $(\omega_1, \omega_2, \dots, \omega_d) \in R^d$ ， $(\omega_{12}, \omega_{13}, \dots, \omega_{1d}, \dots, \omega_{d1}, \omega_{d2}, \dots, \omega_{(d-1)d})$ 是整个模型的参数。特别的，经过关联之后，对于某些特征会提高关联度。

1.2.1.3 混合推荐算法

混合推荐，一般都是通过多个不同推荐算法的结合来生成推荐列表，效果理论上比单一的算法要好很多，但是算法的复杂度较高。

不同的推荐算法往往有着不同的应用场景。有时候，一个算法正好可以弥补另外一个算法的缺陷。对于一些复杂的推荐环境，往往会采用混合推荐模型。通过算法的相互作用，提升推荐效果。对于混合推荐算法而言，一般会有加权、变换、特征组合、层叠、特征扩充等实现形式。

1.2.2 POI 推荐

1.2.2.1 POI 推荐的特点

和传统的推荐系统所解决的问题一样，当面对信息过载的情况时，人们往往无法做出合适的选择。就地理位置而言，当用户在某一个确定的区域里面活动的时候，往往对于周围的地理信息，了解的不够全面，无法从众多的兴趣点中，选择自己想要去的地方，而基于地理位置的推荐，在分析用户的行为轨迹之后，会向用户推荐未来可能会到达的 POI。这一点，可以促进用户的潜在消费，创造巨大的价值。

基于位置的服务（LBS）能够提供位置感知功能。随着定位技术和移动互联网的快速发展，位置数据的利用越来越被大家重视，POI 推荐^[13-16]在近几年一直是一个广泛研究的课题，通过分析用户的历史签到数据，推荐一些用户感兴趣的地点，提高用户体验^[17]。

GPS 轨迹^[18-19]可以直接应用于 LBSN 数据，因为这两种数据类型表现出一定的相似性，但是 LBSN 签到数据表现出一些独特的特性，这些特性与广泛研究的 GPS 轨迹不同，通过分析 Gowalla 数据集中的签到记录，LBSN 数据表现出的特性有：

1) 数据稀疏性：在整个 12 个月期间，只有 10% 的用户拥有 58 个以上的签到记录，用户的签到频率较低。此外，任何两个连续签到之间的空间间隔通常为千米级，而连续记录的 GPS 点之间的空间间隔通常为 5 到 10 米^[18]。

2) 语义性：每个签到记录都用 LBSN 中的位置名称和类别进行标记，而 GPS 点仅由纬度、经度和时间戳组成。签到类别信息反映了用户偏好和行为的异质性。由于这些差异，现有的基于 GPS 轨迹数据的应用技术不能够直接适用于 LBSN

数据。

位置预测中的固有挑战是位置预测空间非常大——LBSN 中可能有数百万个不同的签到位置。因此，很难通过结合用户移动模式、偏好和时间空间信息来建立模型，以直接预测位置并获得满意的效果。针对这一问题，Ye 等人^[20]将问题分解为两个子问题，第一步预测用户下一步活动的类别，第二步给定类别的预测位置。例如，LBSN 可以预估用户的下一个活动是娱乐，然后它可能推荐用户附近的电影院。这种方法的优点在于显著地减少了预测空间，因为只有少数类别，如食物、购物、娱乐等。因此，在此工作中研究的问题是：根据用户签到的位置，预测用户在下一步的活动类别，并预测最有可能的位置。

不同于传统的推荐，基于位置的推荐具有更多属于自己的特点，包括时间上下文，空间上下文，POI 组的特点等，这些特性都会影响到用户的决策。

对于 POI 推荐的特点，主要面临的挑战如下：

1) 用户偏好不明确。对于 POI 推荐，数据集更多的是 LBSN 中签到数据，虽然，LBSN 中存在大量的签到信息，但是，这无法反映用户的真实偏好，用户对某一个 POI 进行签到，只能表示用户曾经的签到事实。而对于记录本身而言，并不代表用户的兴趣偏好。

2) 时间敏感性。对于用户而言，不同的 POI 具有不同的时间特性，对于餐馆而言，用户的就餐时间一般在中午和下午。这个时间段的餐馆，对于用户而言，往往具有更大的吸引力。而对于不同用户而言，同样的 POI，也具有不同的时间特性，如 KTV 这种，学生党因为没有收入的问题，对于 KTV 的选择更多是中午，因为在这个时间段上，商家往往会推出折扣很大的套餐。

3) 转移连续性。对于 POI 而言，每个 POI 都有自己的特性。而不同特性的 POI 之间，往往相互依存，紧密联系。例如，火车站和宾馆之间，往往联系密切，但是，火车站和电影院之间，往往关系微弱。因此，发掘 POI 之间的连续转移特性，也是 POI 推荐面临的巨大挑战。

4) 数据稀疏性。因为 LBSN 中有着大量 POI，而对于用户而言，其签到记录有限，在数据的显示上，就会表现的极其稀疏。尤其是细化到时间段上，用户的签到记录更为稀少^[21]。

5) 实时性。在 POI 的推荐中，用户最需要的是自己在短时间内最可能去的地方的推荐。而不是说，一味的推荐用户喜欢，但是不符合当前情境的 POI。就好像用户喜欢杭州菜，但是在用户吃过晚餐后，还一味的推荐“橘味”这种杭州

风味的餐厅,这显然是不合适的。所以在对于 POI 的推荐中,我们不仅要考量用户的兴趣偏好,同时,还要根据用户所在的上下文情境进行推荐,从而保证推荐的质量。

6) 影响多样性。对于 POI 的选择上,用户往往会被众多的因素所影响,如时间因素,位置因素,社会关系等。特别的,可能单纯因为下雨的原因,用户放弃了外出就餐的打算。

就商圈中的 POI 推荐而言,用户最希望得到的是实时推荐,目前,实时推荐是 POI 推荐的热点和难点之一^[22]。

1.2.2.2 传统 POI 推荐算法

对于 POI 的复杂特性,研究学者根据 POI 特点提出了不同的推荐策略。其中包括融合地理位置、时间因素、用户偏好、社会关系、POI 流行度等。

1) 融合时间因素

Yuan 等人^[23]认为用户对于 POI 的选择,会受到时间的影响。提出将时间因素融入到 POI 推荐中。算法本身将时间进行划分,计算每一个时间段上用户的相似度,然后,以时间段为单位,生成推荐列表。因为依赖协同过滤算法,加上将签到记录按照时间段划分,使得数据变得更为稀疏。

Yao 等人^[24]使用张量分解模型,根据用户的历史签到记录,建立“用户-时间-POI”的三阶张量,通过张量分解,获取用户对于 POI 的偏好信息。但就算法本身而言,只通过用户的签到次数来反映用户偏好,没有考虑到用户的非显式偏好。同样的,Ying 等人^[25]也通过历史签到记录,创建上下文感知的三阶张量,计算用户对于不同类别的 POI 的偏好。

2) 融合地理位置因素

Ye 等人^[26]根据 LBSN 中的地理特性,通过模型来得到地理位置在 POI 推荐中的影响作用。并结合用户偏好和社会关系来生成推荐列表。

Liu 等人^[27]分析地理位置对于用户签到行为的影响,然后通过非负贝叶斯矩阵分解完成对 POI 的推荐。对于地理位置对用户行为的影响,Zhao 等人^[28]利用 POI 之间距离的幂律分布,将待推荐的 POI 和用户的历史签到 POI 集合建立关系,用以获取地理因素对用户行为的影响。

3) 融合社会关系

Gao 等人^[29]就用户的社会关系对于 POI 的推荐展开了详细研究,作者将用

户的社会关系分为了两类，一类是朋友关系，一类是陌生人关系，按照空间上的距离，将小于距离阈值的称作为近距离，大于距离阈值的称作为远距离。研究表明，就社会关系而言，不管是何种类型的朋友关系，其影响都会大于陌生人关系，而在引入距离阈值之后，远距离朋友关系的影响大于近距离朋友关系的影响，而近距离陌生人关系的影响又会大于远距离陌生人关系的影响。结果可能和用户行为相关，就用户而言，用户在 POI 之间的转移可以分为两种情况，一种是在小范围内的周期性活动，用户的大多数签到都发生在该区域内。另外一种为偶尔性的远距离探索。这里，用户在周期性活动中受朋友的影响较小，而对于远距离 POI 的探索上，受朋友的影响较大。针对这种情况，Ference 等人^[30]利用用户的社会关系，结合协同过滤算法为用户进行远距离的 POI 推荐。

4) 融合内容因素

Yang 等人^[31]结合自然语言处理的算法，将用户对于 POI 的评价内容进行语义上的情感分析，然后将内容因素融入到 POI 的推荐当中。这里可以更好的分析用户偏好，而缺点在于使得数据本身变得更加的稀疏。Liu 等人^[32]为 POI 和用户构建模型。通过分析用户的历史签到数据，生成用户特征模型，然后通过比较 POI 特征模型和用户特征模型的相似度，生成推荐列表。这里作者没有考虑到用户对于 POI 选择的场景性，简单的用传统的推荐算法模型处理 POI 推荐问题。

1.2.2.3 连续的 POI 推荐

Cheng 等人^[33]第一次阐述了连续 POI 推荐问题，然后提出用因式分解个性化马尔可夫链（FPMC）来解决连续推荐的问题。Cheng 等人在推荐 POI 的过程中，在 FPMC 算法中加入了地理影响因子，提出 FPMC-LR 模型。

因为在众多的 POI 中，用户通常只会在一段时间内在很小的一部分 POI 上完成签到行为。所以签到数据往往有很大的稀疏性。针对这种情况，He 等人^[34]不仅用马尔可夫链来推断用户在 POI 上的偏好，而且根据用户的签到行为，按照 POI 的类型和签到时间，来将用户进行分组，来解决数据稀疏的问题。Zhao 等人认为^[35]连续的签到行为受时间的影响。因此，他们使用时间因素来衡量连续签到行为的重要性，并相应的提出了一种连续的 POI 推荐方法。

He 等人对用户的连续签到行为进行建模^[36]，采用贝叶斯个性化排序算法对推荐列表进行排序。作者通过构建“用户-POI-POI”的三阶张量，使得缺失值的准确性更高。

Zhao 等人提出 STELLAR (Spatial-Temporal Latent Ranking) 模型^[37], 将时间感知融入到 POI 的推荐中, 作者通过用户的签到记录, 计算出用户的特征矩阵, 时间特性矩阵和 POI 特征张量, 然后综合考虑用户对 POI 的偏好信息, 时间对于用户决策的影响, 还有就是 POI 之间的相互影响, 构建得分函数。

Feng 等人^[38]利用距离嵌入 (Metric Embedding), 将 POI 映射到低维度的欧氏空间中, 用两个 POI 的欧式距离来确定二者的序列关系, 最后综合序列信息和个人喜好, 提出个性化距离嵌入排名算法 (PRME)。此外, Liu 等人^[39]根据 POI 的访问顺序探究 POI 的上下文信息。首先, 他们利用原本为自然语言处理 (NLP) 设计的 Skip-Gram 模型^[40], 探究 POI 访问顺序的影响。然后, 基于 Weston 提出的个性化推荐模型^[41], 将访问频率带入模型。虽然 Skip-gram 最开始是用于语言处理, 但是算法在连续的 POI 推荐中也有很好的表现。Feng 等人^[42]也利用了单词嵌入技术, 通过用地理二进制树替换原来的霍夫曼树, 修改 Continuous Bag-of-Word (CBOW) 模型来适应地理环境因素。

1.2.3 现有研究存在的问题

传统意义上的 POI 推荐, 基本是静态 POI 推荐, 基本思想是在传统的推荐算法上加上 POI 的特性, 包括时间上下文, 地理位置影响, 用户社会关系, POI 主题因素等。就 POI 推荐而言, 近年来, POI 的实时推荐得到了越来越多的关注, 其关键点在于 POI 之间连续转移的研究。

对于 POI 的研究中, 目前还没有利用连续 POI 推荐思想, 用之于商圈这种特定场景。商圈中的 POI 不同于一般的 POI, 其聚集性和稠密性更高, 用户在商圈中不同 POI 之间的转移也更为频繁, 商圈中的 POI 也更具有特征性。特别是 O2O 商业模式下, 如何有效的利用用户的签到行为, 根据用户的当前位置, 为用户推荐下一个更值得去的地方, 是一个值得被讨论的问题。

1.3 研究内容和技术路线

本文的研究内容是商圈中的连续 POI 推荐。通过用户的签到行为, 获取用户的实时位置, 结合 POI 之间的连续过度影响, 为用户生成实时的推荐列表, 提升用户在商圈中的用户体验。本文的研究内容如下:

- 1) 本文详细介绍了几种常见的推荐算法, 并且根据推荐算法在各行各业中

的应用，提出商圈中的 POI 推荐问题，根据 POI 特性，并结合商圈中 POI 的聚集性和连续过度性，提出采用连续推荐处理商圈中的连续 POI 推荐问题。

2) 实验比较了 UGSE-LR 和 PEU-RNN 算法^[17]在连续推荐上的性能问题，UGSE-LR 考虑了用户偏好，区域影响，还有 POI 之间的连续过度影响。适合中等规模的数据集。而 PEU-RNN 基于神经网络，需要大量的数据来发掘用户偏好和 POI 之间的连续过度影响，适合大规模数据集。

3) 针对商圈特性，整理功能需求，结合 UGSE-LR 模型，进行了商圈中 POI 连续推荐的系统开发。引入评分机制，解决用户偏好不明确问题。并引入标签体系，解决系统冷启动问题。

本文的技术路线如图 1-4 所示：

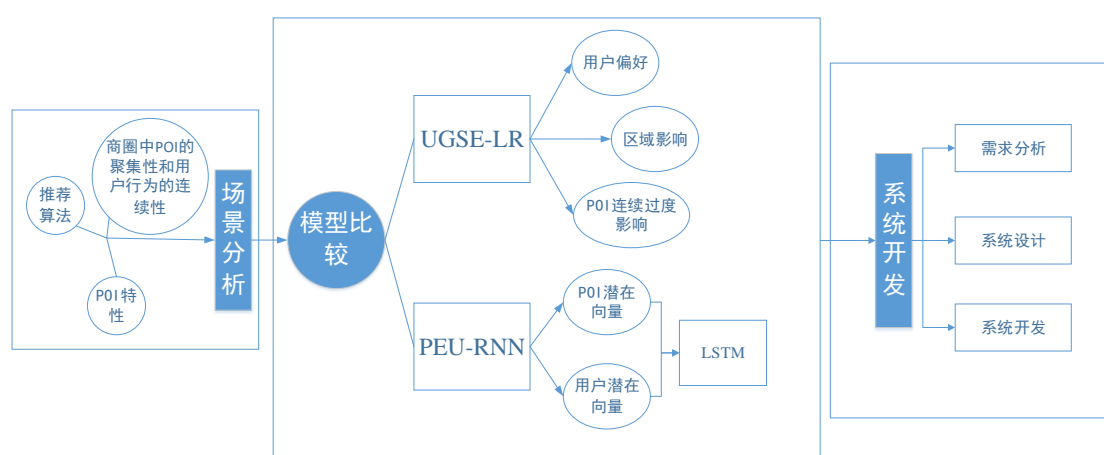


图 1-4 技术路线图

1.4 论文的组织结构

本文第一章主要是对研究方向进行了相关的调研工作，明确了研究意义，了解了推荐系统的研究现状，并针对近年来出现的连续 POI 推荐，做出了比较详细的介绍。

第二章主要是详细介绍了 UGSE-LR 模型，根据连续 POI 推荐的思想，将 POI 和用户的签到数据映射成有向图中的顶点和有向边，采用个性化 PageRank 处理 POI 到 POI 的连续过度影响。然后根据用户的签到位置设置网格，计算区域影响，最后引入用户的偏好信息计算 POI 得分，将得分最高的 POI 推荐给用户，并通过准确率和召回率来确定模型的参数。

第三章主要是介绍了 PEU-RNN 模型。模型借鉴自然语言处理的思路,利用 Continuous Bag-of-Word (CBOW) 对用户和 POI 进行编码,然后将 POI 和用户的潜在向量作为 LSTM 的输入,构建模型,预测下一个高概率访问的 POI。实验表明,在大量数据集的训练下,模型有着较好的推荐效果。

第四章,根据商圈中的 POI 特性,选择 UGSE-LR 模型作为处理 POI 的连续推荐,并引入标签体系,解决系统的冷启动问题,完成商圈中连续 POI 推荐系统的设计和实现。

第五章主要是对自己工作的总结以及展望。

第 2 章 基于个性化 PageRank 的 POI 连续推荐

2.1 问题定义

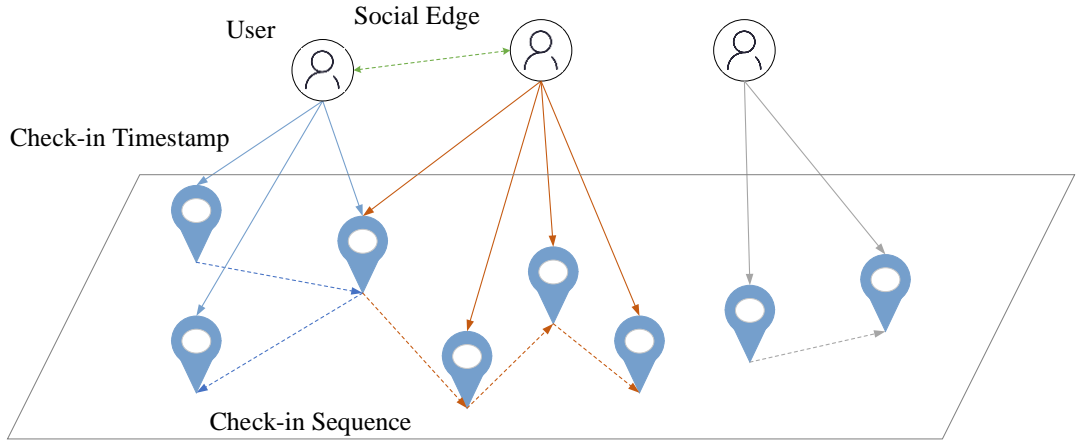


图 2-1 LBSNs 中的签到表示

如图 2-1 所示，基于位置的社会网络（LBSNs）的签到记录可以定义如下：

U 和 L 分别代表用户和 POI 的集合， L_u 表示用户 u 已经访问过的 POI 集合，对于给定的 $q(u, l_c, t_c)$ ，其中 $u \in U$ ，为目标用户， $l_c \in L_u$ 是用户的签到地点， t_c 是签到时间，连续 POI 推荐的问题在于向用户 u 推荐 N 个即将想去的 POI，可以用式子表示为 $R_{u,N}$ 。其中对于每一个 POI， $l \in R_{u,N}$ ，满足下面条件：

- 1) l 和 l_c 之间的距离要小于等于 d 。
- 2) 用户 u 会在时间段 $[t_c, t_c + \tau]$ 内，在 l 处完成签到行为。

2.2 数据集描述

本次实验的数据集为 Gowalla 和 Brightkite，为了更好的反映模型的效果，我们对数据集进行预处理，分别去掉那些不受欢迎的冷门 POI 和一些不活跃的用户。具体的去除标准如下：

- 1) 少于 80 个签到记录的用户签到记录。
- 2) POI 中，少于五个用户签到的位置。
- 3) 用户中少于五个好友的签到记录。

通过数据清理，数据集中的数据表示如下：

表 2-1 数据集记录

数据库	Brightkite	Gowalla
用户	3178（人）	6415（人）
POI	6792（个）	44938（个）
签到记录	291655（条）	819283（条）
社会关系	45308（条）	73410（条）
时间范围	2008.04-2010.10	2009.02-2010.10

如图 2-2 所示，对于 POI 的连续签到而言，用户的签到行为在很大程度上，会受到上一时间段签到行为的影响，图中显示的是 Gowalla 和 Brightkite 中两个连续签到记录的时间差在所有签到记录中的比例。在 Gowalla 中，有 52% 的连续签到行为在 6 个小时内，72% 的记录发生在 24h 内，在 Brightkite 中，有 40% 的连续签到行为在 6 小时内，60% 的记录发生在一天内。这里可以看出，Brightkite 数据集在时间域上比 Gowalla 数据集更加的稀疏。

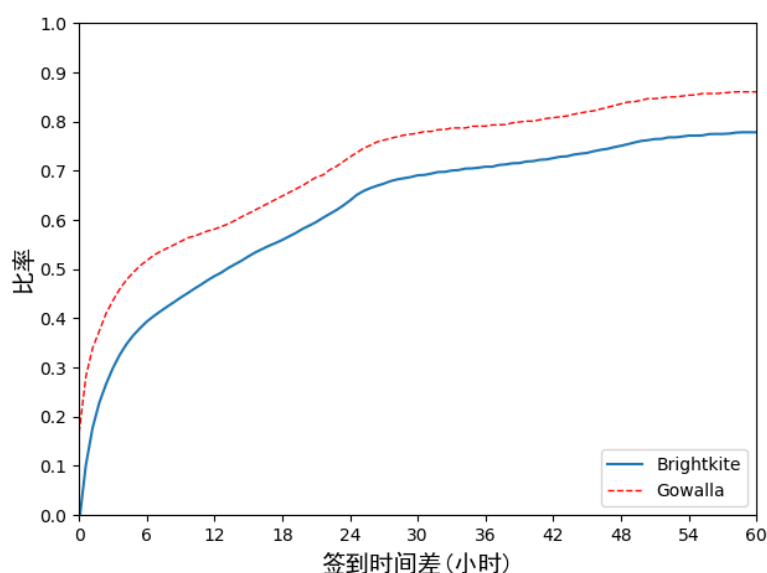


图 2-2 签到时间差比率图

除了时间差异之外，POI 之间的距离也会影响用户决策。图 2-3 是连续签到行为发生在 6 小时内，签到地点距离差的分布图。从图中可以看到，在 Gowalla 中，超过 90% 的连续签到记录在距离当前位置 15km 的范围内，而在 Brightkite 中，也有超过 50% 的数据。很显然，Brightkite 数据集在空间领域也比 Gowalla 数据集更稀疏。

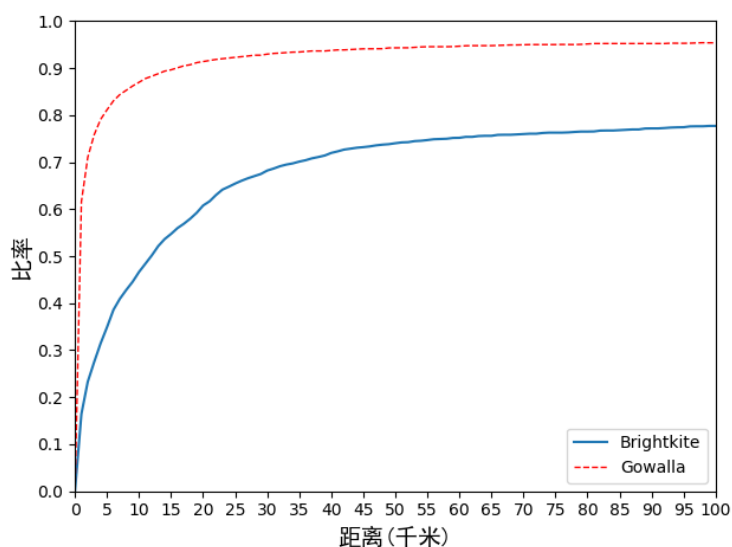


图 2-3 距离比率图

2.3 模型架构

对于用户而言，在 POI 的选择上，往往会受到偏好和当前位置的影响，如图 2-4 所示，我们根据历史签到记录和用户的当前位置，将用户偏好、区域影响、以及 POI 的连续过度影响作为影响因子，为用户生成推荐列表。

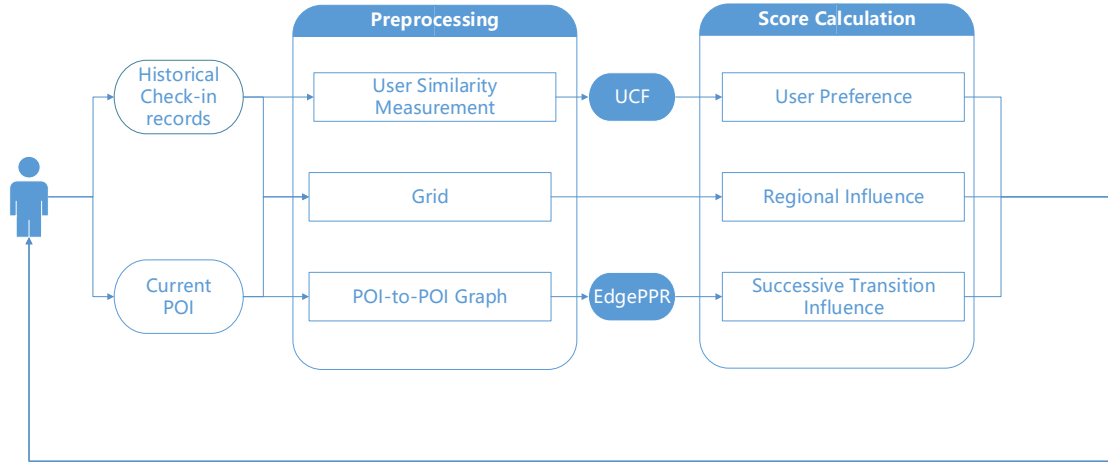


图 2-4 UGSE-LR 模型架构图

2.3.1 用户偏好影响

对于用户偏好的计算，计算方法如下：

对于用户 v 和 u ，让 $C_{u,l} = 1$ 表示用户 u 已经在 POI l 处进行过签到行为，否则 $C_{u,l} = 0$ ，根据用户 v 和 u 的历史签到记录，用户 u 和用户 v 的相似度可以用下面的等式表示：

$$\omega_{u,v} = \frac{\sum_{l \in L} C_{u,l} \cdot C_{v,l}}{\sqrt{\sum_{l \in L} C_{u,l}^2} \sqrt{\sum_{l \in L} C_{v,l}^2}} \quad (2-1)$$

U' 是用户 u 在签到行为上相似的用户的集合，用户 u 对 POI l 偏好程度的分数可以用下面的等式表示：

$$p_{u,l}^{user} = \frac{\sum_{v \in U'} \omega_{u,v} \cdot C_{v,l}}{\sum_{v \in U'} \omega_{u,v}} \quad (2-2)$$

2.3.2 区域影响

与传统的 POI 推荐不一样，连续 POI 推荐很大程度上会考虑推荐 POI 和当前用户所在位置的距离。如图 2-3 中的数据显示，用户选择下一个 POI 时，很大程度上会选择距离自己较近的。这也就意味着，用户很少会选择一个比较有趣的，但是距离自己现在位置很远的地方。因此，我们可以将距离远近作为一个影响因子，放在连续的 POI 推荐中，如图 2-5 所示，我们将地理位置划分为很多大小相等的格子，以用户 u 当前位置 l_c 为圆心，阈值 d 为半径作圆。圆与网格相交的阴影

部分，称之为用户 u 的临近网格。

这里网格的划分来源于生活实际，一个区域内，有很多人进行签到行为，那么这个区域必定是一个受欢迎的区域，与之相对应的，这个区域里面的 POI，也会更受欢迎。就好像每个热门商圈中的店铺一样，用户在 POI 之间的转移往往只会在商圈内部。很少会因为某一个 POI，而离开当前商圈。特别的，同一商圈中的 POI，往往会有相同的特性。

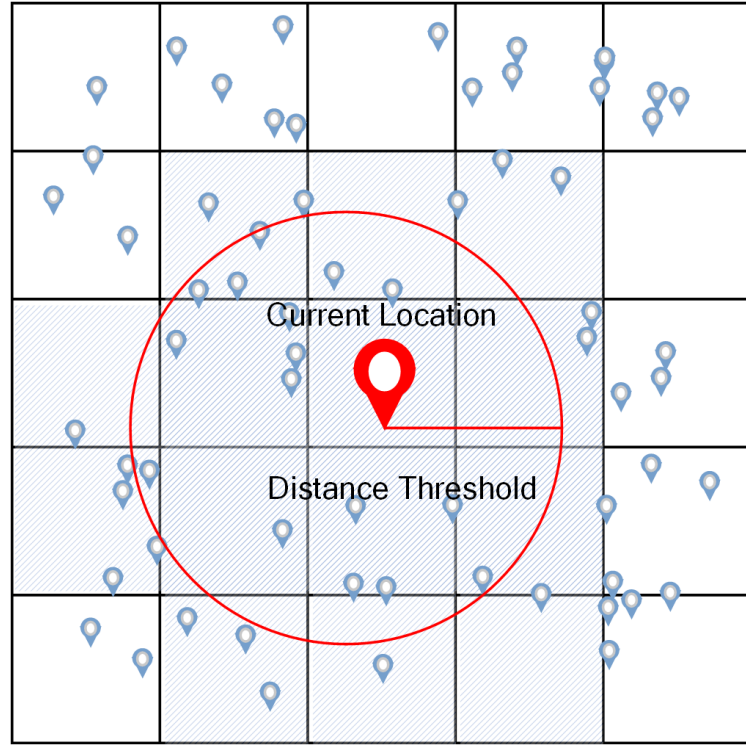


图 2-5 区域的网格划分

这里我们让 $Checkins(g_i)$ 作为网格单元 g_i 中所有签到记录的总和，那么网格 g_i 的受欢迎程度可以用等式表示为：

$$g_i^p = \frac{Checkins(g_i)}{\sum_{g \in G_s} Checkins(g)} \quad (2-3)$$

其中 G_s 是用户 u 的临近网格单元的集合。

当用户 u 在某个网格单元 g_i 中，有很多的签到行为。那么可以说 g_i 区域是用户 u 最喜欢的区域，用户 u 有很大的可能会在这个 g_i 网格中的其他 POI 上进行签到行为。就好像用户经常活动以及发生消费的位置，都会是某个固定的商圈。这

里我们让 $Checkins(g_i, u)$ 表示用户 u 在网格 g_i 中所有 POI 的签到记录总和，我们可以用如下公式来表示这种影响：

$$g_i^u = \frac{Checkins(g_i, u)}{\sum_{g \in G_S} Checkins(g, u)} \quad (2-4)$$

正如图 2-3 所示，用户往往倾向于在距离当前位置较近的 POI 上发生签到行为，因此，我们可以推断出，用户倾向于在当前位置所在的网格内的其他 POI 上进行签到行为，我们可以将这种影响表示为下面的等式：

$$g_i^c = \begin{cases} 1 & \text{如果当前用户所在的位置在 } g_i \\ 0 & \text{其他情况} \end{cases} \quad (2-5)$$

我们将三种影响因子混合在一起，通过线性方程组将网格 g_i 得分表示为如下等式：

$$GridScore_{g_i} = \alpha g_i^p + \beta g_i^u + \gamma g_i^c \quad (2-6)$$

其中 α 、 β 还有 γ 是满足下列条件的常量。

$$\begin{cases} 0 \leq \alpha, \beta, \gamma \leq 1 \\ \alpha + \beta + \gamma = 1 \end{cases} \quad (2-7)$$

最后 POI l 对于用户 u 的影响，可以用下面的等式表示：

$$P_{u,l}^{reg} = \frac{GridScore_{g_i}}{\sum_{g \in G_S} GridScore_g} \quad (2-8)$$

其中 g_i 是 POI l 所在的网格。

2.3.3 连续过度影响

我们用 POI 到 POI 的转移图来建模签到记录中的连续签到行为，用 (l, t) 来表示用户在时间点 t 上，在 POI l 上进行了签到行为，那么 POI 到 POI 的转移图可以定义如下：

定义 1：用户 u 的一系列签到行为 $(l_1, t_1), (l_2, t_2), \dots, (l_n, t_n)$ ，其中 $t_1 \leq t_2 \leq \dots \leq t_n$ ，如果说 $t_n - t_1 \leq \tau$ ，说明 POI l_i 和 l_{i+1} 之间有连续过度影响。

定义 2：POI 到 POI 的过度图，可以表示为有向图 $G = (L, E)$ ，其中 L 是 POI

的集合, E 是 L 中所有具有连续过度影响的边。也就是说, 如果在所有用户的历史签到记录中, 从 l_i 到 l_j 有连续过度影响, 那么就存在一条有向边 (l_i, l_j) , 这个地方, (l_i, l_j) 边的权重可以定义为:

$$w_{l_i, l_j} = \frac{Transitions(l_i, l_j)}{\sum_{l \in L} Transitions(l_i, l)} \quad (2-9)$$

这里 $Transitions(l_i, l_j)$ 是所有用户签到记录中, l_i 到 l_j 的连续过度记录的总数。

因为考虑到距离阈值, 被推荐给用户的 POI, 一定是在用户 u 的临近网格中, 我们可以从全局 G 中, 创建子图 $G' = (L', E')$, 从全局 G 中去除掉不在 u 临近网格中的 POI, 然后再用 Edge-weighted Personalized PageRank(EdgePPR)^[43] 来计算子图中所有结点的权重, 用户 u 到 l 的连续过度影响可以表示为下面的等式:

$$P_{u, l}^{suc} = \frac{EdgePPR(G', l)}{\sum_{l' \in L'} EdgePPR(G', l')} \quad (2-10)$$

然后利用最大最小归一化方法, 对用户偏好, 区域影响和连续过度影响进行归一处理。

$$\begin{aligned} S_{u, l}^{user} &= \frac{P_{u, l}^{user} - m_{u, l}^{user}}{M_u^{user} - m_u^{user}} \\ S_{u, l}^{reg} &= \frac{P_{u, l}^{reg} - m_{u, l}^{reg}}{M_u^{reg} - m_u^{reg}} \\ S_{u, l}^{suc} &= \frac{P_{u, l}^{suc} - m_{u, l}^{suc}}{M_u^{suc} - m_u^{suc}} \end{aligned} \quad (2-11)$$

这里 M_u^{user}/m_u^{user} , M_u^{reg}/m_u^{reg} , M_u^{suc}/m_u^{suc} , 分别是在区域 L' 中, 用户偏好, 区域影响, 还有连续过度影响的最大最小值。对于用户 u 而言, POI l 的得分可以用下面的式子表示:

$$S_{u, l} = \delta \times S_{u, l}^{user} + \varepsilon \times S_{u, l}^{reg} + \xi \times S_{u, l}^{suc} \quad (2-12)$$

其中 δ , ε , ξ 是满足下面条件的常量。

$$\begin{cases} 0 \leq \delta, \varepsilon, \xi \leq 1 \\ \delta + \varepsilon + \xi = 1 \end{cases} \quad (2-13)$$

2.4 POI 之间连续过度影响建模

PageRank 算法本身最初被用来处理网页排名^[44]，但是该方法现在被广泛应用于各种应用中，如对象数据库、社交网络和推荐系统等^[45-48]。

2.4.1 全局 PageRank

PageRank 算法本身因为应用在商用搜索软件 Google 上，而获得巨大成功。虽然网页的数量巨大，但是用户在网页搜索的时候，并没有感受到时间的差异性，这是因为 PageRank 算法是离线计算。传统的 PageRank 和查询条件无关，也被称为全局 PageRank，可以在离线的情况下得出所有网页的权重。

对于全局 PageRank 而言，page 等人将网页之间的链接关系表示成有向图 $G = \langle V, E \rangle$ 。其中，结点 $v \in V$ 表示网页，有向边 $e = (v_i, v_j)$ 表示网页之间的链接关系。利用随机游走模型来模拟用户在网页之间的跳转，用户从任意的结点开始，以 α 的可能性到达下一个可达结点，或者以 $1 - \alpha$ 的可能性开始新一轮的随机游走， α 的取值一般为 0.15。用户不断迭代这种行为，直到对于有向图而言，用户停留到每个网页的可能性趋于稳定。可以用如下公式表示：

$$r = (1 - \alpha)Mr + \alpha u \quad (2-14)$$

其中， α 为跳转到下一结点的可能性， M 为有向图 G 的邻接矩阵， u 为均值向量。

2.4.2 个性化 PageRank

而对于 PageRank 的扩展算法。个性化 PageRank 更多是被当作一种排序算法，其应用场合更多的是可以表示成图结构的数据集。利用图中结点之间的链接联系，通过递归计算，得出每一个结点的权重。但是对于个性化 PageRank 而言，不仅需要考虑结点间的链接关系，而且还要添加个性化信息。所以对于计算效率而言，是一个很大的挑战。

通过个性化 PageRank 得到的结点权重，体现的是结点和搜索条件之间的联系程度，称之为个性化 PageRank 向量，表示为 PPV (Personalized PageRank Vector)^[49]。对于数据集规模比较大的情况下，PPV 的计算往往需要很大的计算量。特

别是对于实时查询，用户对响应时间有着严格的要求。而对于采用离线计算的方式存储所有的可能来缩减用户的等待时间，这对于系统而言，显然是一件不可能的事情。Fogarás 等人^[50]已经证明，对于 n 个结点，需要的存储空间至少是 $\Omega(n)^2$ 。这也就是说，用户偏好可能是 $2n - 1$ 个任意子集。

对于个性化 PageRank 而言，与传统 PageRank 的差异在于随机游走过程里面的跳转行为，用户在进行跳转的过程中，只能跳转到用户偏爱的部分结点，所以，用户偏爱的结点往往能得到更高的权重分配，个性化 PageRank 可以用公式表示如下：

$$r = (1 - \alpha)Mr + \alpha v \quad (2-15)$$

其中 v 表示用户的个性化向量， $|v| = 1$ ，如果用户偏好的结点有 k 个，那么这 k 个结点的值之和为 1，其他结点的值为零，式子的解就是个性化向量 v 对应的 PPV。

对于个性化 PageRank，可以调整结点权重或边权重，以确定随机冲浪者模型中的传送概率和转换概率。然而，自十多年前个性化 PageRank 出现以来，计算效率一直是一个很大的问题。Xie^[43]提出边权个性化 PageRank (Edge-Weighted Personalized PageRank)，使得个性化推荐的性能得到了很大的提高。本章第三节中的连续过度影响，就是将用户在 POI 之间的转移进行模型化处理，将 POI 之间的路径关系表示成一个有向图 $G = \langle V, E \rangle$ 。其中，结点 $v \in V$ 表示 POI，有向边 $e = (v_i, v_j)$ 表示历史记录中 POI 之间的连续签到。将用户在 POI 之间的转移看作是一条马尔可夫链，也就是说，用户对下一个位置的选择，只和当前位置有关，然后，用边权个性化 PageRank 计算 POI 的个性化 PageRank 向量。其中，权重较高的 POI 表示对于用户而言从当前位置到目标 POI 的连续过度影响较大。

2.5 实验设计与验证

2.5.1 实验设计

准确率和召回率被广泛用于评价推荐系统的性能，所以，在本次实验中，分别用 Precision@N 和 Recall@N 来表示实验的准确率和召回率，Precision@N 和 Recall@N 可以用如下的等式表示：

$$Precision@N = \frac{|R_{u,N} \cap L'_{u,l_c,t_c}|}{N} \quad (2-16)$$

$$Recall@N = \frac{|R_{u,N} \cap L'_{u,l_c,t_c}|}{|L'_{u,l_c,t_c}|} \quad (2-17)$$

上式中 $R_{u,N}$ 是 top-N 中的 POI 集合， l_c 是用户当前位置所在的 POI。 L'_{u,l_c,t_c} 是用户在 POI l_c 的位置上，在时间段 $[t_c, t_c + \tau]$ 内，访问的 POI 的集合。

为了评估 UGSE-LR 和 PEU-RNN 方法的性能，我们与前人的 FPMC^[51]、FPMC-LR^[33]、POI2VEC^[42]方法做比较。方法的特点如表 2-2 所示。

表 2-2 方法比较

方法	用户偏好	地理影响		连续过度影响
		距离影响	区域影响	
FPMC	√			√
FPMC-LR	√	√		√
POI2VEC	√		√	√
UGSE-LR	√	√	√	√
PEU-RNN	√	√		√

2.5.2 参数的确定

在方法中有两组参数需要确定，分别是公式（2-6）中的 $\{\alpha, \beta, \gamma\}$ ，还有公式（2-12）中的 $\{\delta, \varepsilon, \xi\}$ 。Gowalla 和 Brightkite 中的最优参数如表 2-2 所示：

表 2-2 UGSE-LR 模型参数确定

数据集	α	β	γ	δ	ε	ξ
Gowalla	0.3	0.2	0.5	0.2	0.3	0.5
Brightkite	0.2	0.2	0.6	0.5	0.3	0.2

由表数据可以得出，无论是 Gowalla 还是 Brightkite 数据集， γ 的值都是最大的，说明影响签到行为的主要因素是用户的当前位置。这一点也和实际的生活一致。对于用户偏好、区域影响和连续过度影响，从表中可以看出，对于 Gowalla 数据集，连续过度影响是最重要的影响因素。而在 Brightkite 数据集中，用户偏好是影响访问意愿的关键。我们认为这种差异可能是由用户行为引起的。在

Gowalla 系统中, 有一个行程推荐系统, 让用户更容易跟随行程建议。因此, 连续过度影响在 Gowalla 数据集中起着重要的作用。

从数据集本身来说, 由图 2-2 和图 2-3 可以看出, 在 Brightkite 数据集中, 两次连续签到的距离和时间差比 Gowalla 数据集中的距离和时间差要长。这里和数据的收集有关。在 Brightkite 系统中, 允许用户自由的在任何 POI 上进行签到, 这里的 POI 可能是用户在短时间上无法到达的地点, 这种机制使得用户偏好成为 Brightkite 数据集上的关键影响。

2.5.3 网格大小和距离阈值的影响

对于区域影响因子的确定, 我们用网格对地图进行了划分, 然后根据距离阈值作圆确定临近网格单元。为了确定网格范围和距离阈值 d 对于模型的影响, τ 值被设置为两个小时。推荐给用户的 POI 的数量 N 被设置为 10, 网格大小分别设置为 0.2km、0.5km、5km 和 20km, 距离阈值 d 设置为 0.5km、1km、5km、10km、50km、100km。实验结果如图 2-6、2-7 所示:

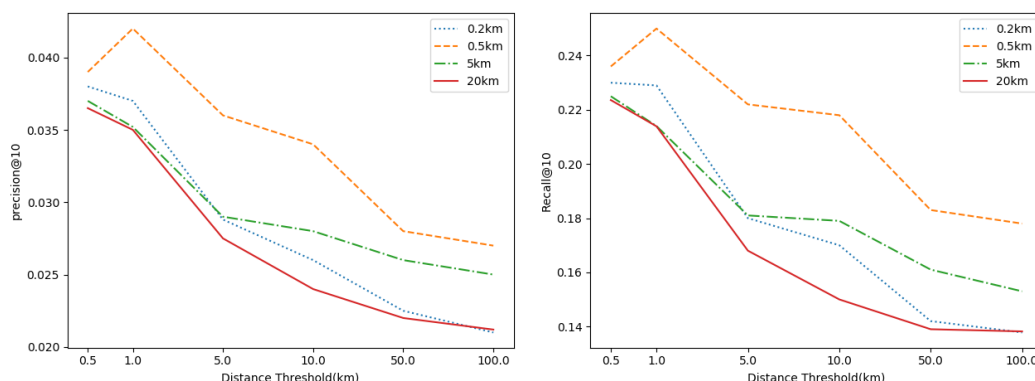
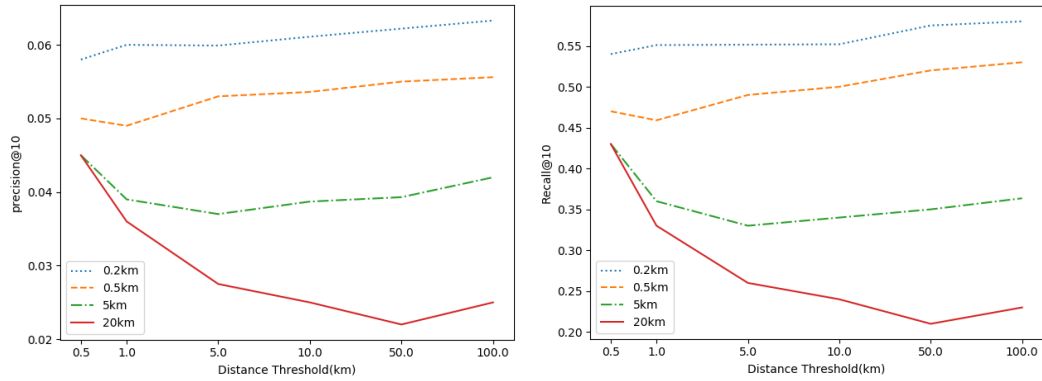


图 2-6 Gowalla 中不同 d 值对应的准确率和召回率


 图 2-7 Brightkite 中不同 d 值对应的准确率和召回率

从图 2-6 可以看出, 在 Gowalla 数据集上, 当阈值设置为 1km 时, 可以得到最好的推荐结果, 然后随着距离阈值的增大, 模型的效果发生下降。这意味着如果距离阈值设置过大的时候, 模型需要考虑更多的 POI 作为候选 POI, 因为用户通常会在当前位置的附近发生下一次的签到行为, 所以更多的候选 POI 使得推荐更具有挑战性。而对于图 2-7, 在 Brightkite 数据集中, 却发生了一个有趣的现象, 随着距离阈值的增加, 推荐的效果反而更好。这个地方还是和数据集本身的特性有关。因为 Brightkite 系统可以直接完成签到行为, 而不用到达 POI 后完成签到行为。这也就导致了用户在选择下一个 POI 时, 很小的程度上会考虑过远的实际距离。

网格大小对区域影响非常重要。如果网格大小设置过大, 那么每个网格中的 POI 的数量就会增加。导致区域影响占据过大的权重。另一方面, 如果网格设置的过小, 那么, 每个网格中的 POI 数量就会减少。这就会导致网格缺乏区域特性。由图中数据可以看出 Gowalla 数据集中, 当网格大小设置为 0.5km 时, 推荐效果最好, 而对于 Brightkite 数据集来说。网格大小设置为 0.2km, 可以得到最好的推荐效果。

2.5.4 性能比较

2.5.4.1 推荐列表中 POI 数量的影响

对于模型中的网格大小设置, Gowalla 数据集被设置为 0.5km, 在 Brightkite 数据集中, 被设置为 0.2km。推荐列表中 POI 的数量 N 的值分别设置为 5、10、15、20、25。由图 2-8、2-9 所示, UGSE-LR 的性能优于 FPMC, FPMC-LR 和

POI2VEC。原因是模型不仅考虑了 POI 与用户之间的距离，而且还包括 POI 所在区域对用户决策行为的影响。

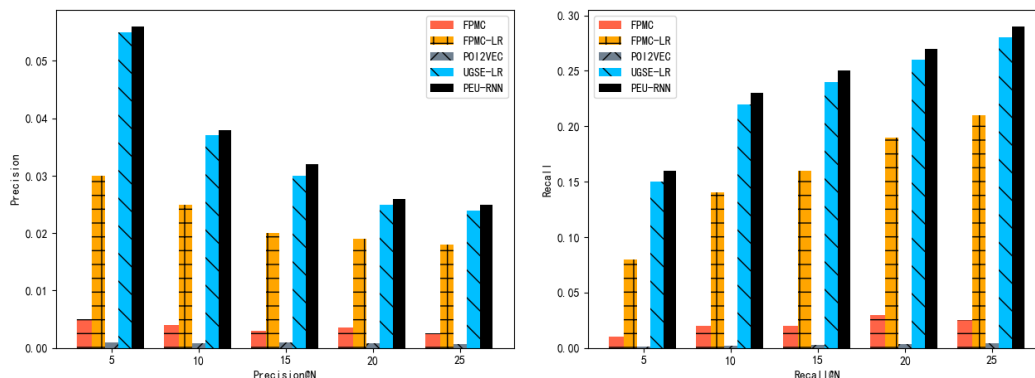


图 2-8 Gowalla 数据集中的准确率和召回率

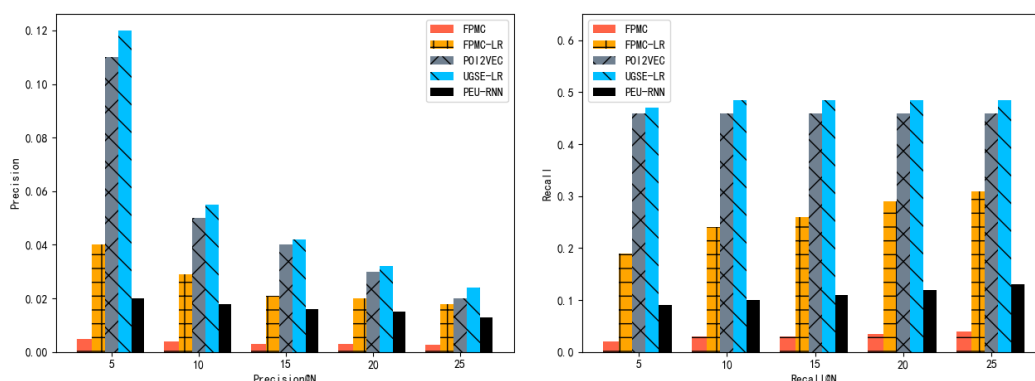


图 2-9 Brightkite 数据集中的准确率和召回率

图 2-10 是 Gowalla 和 Brightkite 数据集中，在 τ 值分别取 3h 和 6h 时，关于连续签到的数量分布。在 Brightkite 的训练数据集中，有百分之九十的记录在三小时内只完成了一次签到，但是在 Gowalla 数据集中，超过一次的连续签到行为的数据超过百分之三十。

正如之前所提到的，在 Brightkite 数据集中，用户偏好的影响是最大的，连续转变的影响较低。UGSE-LR 模型因为考虑到用户偏好，区域影响还有连续转变影响的原因，在大多数情况下仍然优于其他方法。在 Gowalla 数据集的对比实验中，我们发现 POI2VEC 的性能是最差的，这和 Feng^[42]的实验结果相冲突，但是在 Brightkite 数据集中却具有良好的效果，特别是在 N 比较小的情况下。我们认为在 Gowalla 数据集中的实验结果是由于进行实验的 Gowalla 数据集的规

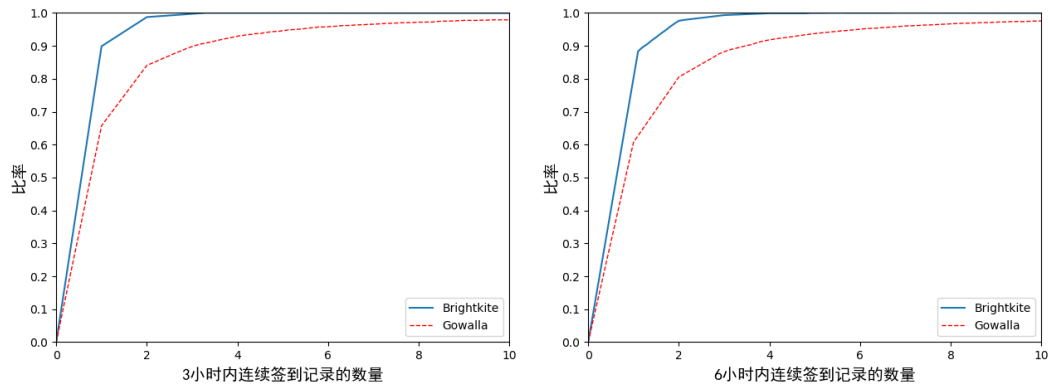


图 2-10 不同时间段连续签到的比例分布

模所导致的。在 Feng 的实验中^[42]中，只使用了部分 Gowalla 数据集（准确来说只有一个城市），而在我们的研究中，使用了全部 Gowalla 数据集。在 Brightkite 数据集中，随着推荐 POI 数量的增加，POI2VEC 的性能逐渐降低。这是因为 POI2VEC 不考虑距离约束，导致算法推荐的 POI 距离用户很远。而这与实际的生活相违背。不同的是，由于考虑了多种因素，当 POI 数量增加时，UGSE-LR 总能表现良好，并且推荐效果稳定。

2.5.4.2 时间约束 τ 的影响

对于时间阈值 τ 的确定，我们从 1 到 6 小时进行取值，实验结果如图 2-11 所示：

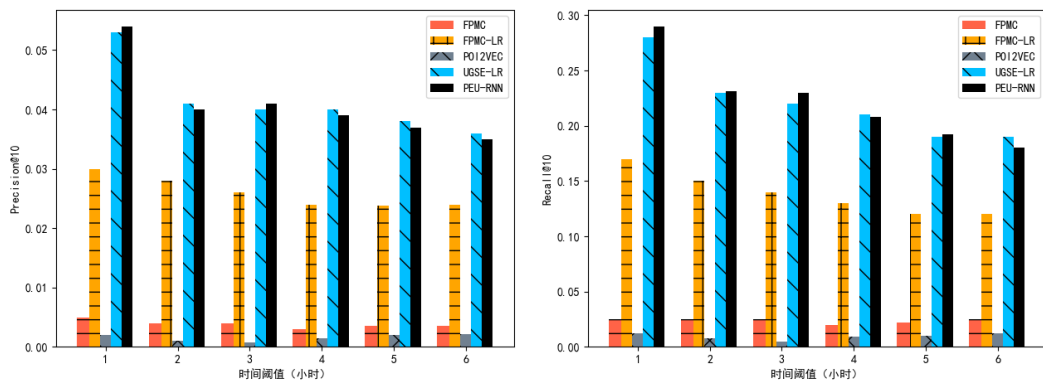


图 2-11 Gowalla 数据集中的准确率和召回率

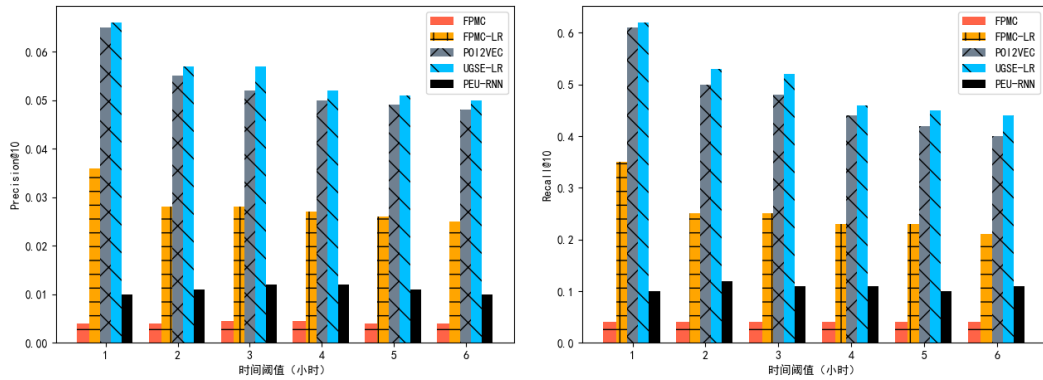


图 2-12 Brightkite 数据集中的准确率和召回率

由图 2-11、2-12 所示，无论是 Gowalla 数据集还是 Brightkite 数据集，随着时间 τ 值的增加，UGSE-LR 模型的推荐效果都发生了降低。但是无论是那种情况下，与其它的推荐算法相比，效果都是比较好的。

2.6 本章小结

本章主要是对连续推荐进行了定义，并根据连续推荐的特点，预处理了数据集。针对商圈中 POI 的特性，采用 UGSE-LR 模型处理 POI 的推荐问题。

针对 UGSE-LR 模型中的用户偏好，区域影响，连续过度影响。通过 Precision@N 和 Recall@N，确定了不同数据集下的最优参数。并通过对比实验，验证了模型的推荐效果。

第3章 基于神经网络的 POI 连续推荐

POI连续推荐的思想在于发掘用户在签到行为上的序列关系,这一点和RNN(Recurrent Neural Network)解决的问题一致。利用词嵌入技术,可以将连续的POI推荐看作是序列预测问题。但是传统的编码方法,像 one-hot 编码不能包含前一个访问的 POI 和下一个访问的 POI 之间的隐式关系,因此,算法利用 Mikolov^[40]提到的 Continuous Bag-of-Word(CBOW),通过用户的历史签到记录,对用户和 POI 进行编码。

根据用户偏好和 POI 之间的连续过度影响,CBOW 能够为每个用户和 POI 提供一个独特的潜在向量,这些潜在向量能够用来表示 POI 的连续性和用户的偏好信息。然后,模型将 POI 和用户的潜在向量作为 RNN 的输入,构建基于 RNN 的模型,预测下一个高概率访问的 POI。

由于图形处理单元(GPU)的显著进步,现代神经网络模型如卷积神经网络(CNN)和递归神经网络(RNN),可以利用大量的隐含层提取有用的预测因子。目前 CNN 广泛应用于图像处理、手写识别等。然而,CNN 在序列数据上表现不佳。相反,RNN 可以将之前的结果依次考虑到当前阶段,从而使 RNN 具有“记忆”的能力。因此,RNN 通常适用于序列数据,如序列点击预测、语言模型、语音识别等。然而,当序列太长时,RNN 仍然存在梯度爆炸或梯度消失等问题,前面的结果可能无法保留到后面的输入。为了解决这些问题,RNN 出现了两个变体,分别是 Long Short-Term Memory(LSTM)和 Gated Recurrent Unit(GRU)。

由于 POI 签到记录的特点通常包含较长的顺序签到,因此在模型 PEU-RNN 中采用 LSTM 作为预测模型。

3.1 模型架构

PEU-RNN 模型架构如图 3-1 所示,第一阶段,可以从用户的历史签到记录中获取到所有用户和 POI 的潜在向量,潜在向量包含了用户的访问偏好和 POI 的连续过度影响。首先,我们利用所有用户在一段时间内的连续签到记录通过分

布式表示找到 POI 之间的转移关系，然后使用同样的方法，根据用户的签到记录，从其 POI 的访问序列中，构建代表用户的潜在向量。第二阶段，将历史签到记录和第一阶段获得的潜在向量构建 POI 推荐的 RNN 模型，利用 softmax 函数作为激活函数，根据用户的当前位置和已访问的 POI，计算下一个 POI 的访问概率。

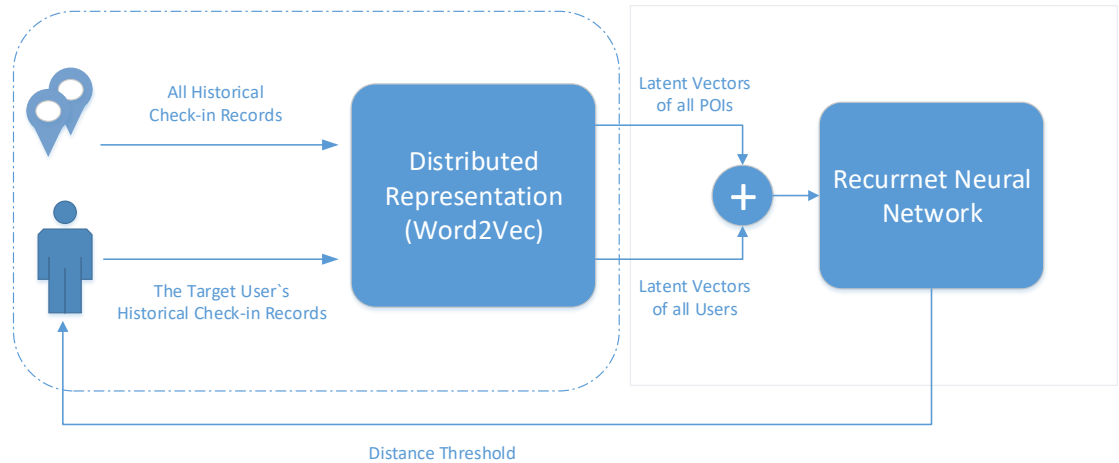


图 3-1 PEU-RNN 模型架构

3.2 潜在向量的建立

POI 之间的连续转移对于 POI 推荐非常重要，但是，UGSE-LR 采用的 POI-to-POI 转移图过于复杂，不适合 RNN。基于 RNN 在 NLP 中的显著成功，每个 POI-to-POI 的转移都应该基于向量编码方法表示成一个潜在向量。

现在流行的矢量编码是 one-hot 编码和分布式表示。当使用 one-hot 编码时，POI 之间的连续转移将转化为高维向量，从而带来巨大的计算开销。使用分布式表示时，可以通过将数据维数降低到可接受的范围来控制计算开销。此外，分布式表示还考虑了前后 POI 的序列关系，因此，我们认为分布式表示比 one-hot 编码更适合 POI 推荐。

word2vec 技术由两个模型组成，分别是 CBOW 和 Skip-gram。这两种都是用霍夫曼树来建立上下文关系。CBOW 可以根据当前的 POI 预测下一个 POI。然而，Skip-gram 模型可以根据当前的 POI 来预测先前 POI 和下一个 POI。由于 POI 连续推荐的特点，在模型中采用 CBOW 对 POI 之间的连续转移进行建模。

具体流程如下：每个用户在时间 τ 范围内的签到被表示成一个序列 $L_i = (l_1, \dots, l_m | \tau \geq (t_m - t_1))$, $\{l_j | 1 \leq j \leq m, l_j \in L\}$ 。 L_i 包含 POI 的访问序列和频率，可以在 CBOW 中构建霍夫曼树，如图 3-2 所示：

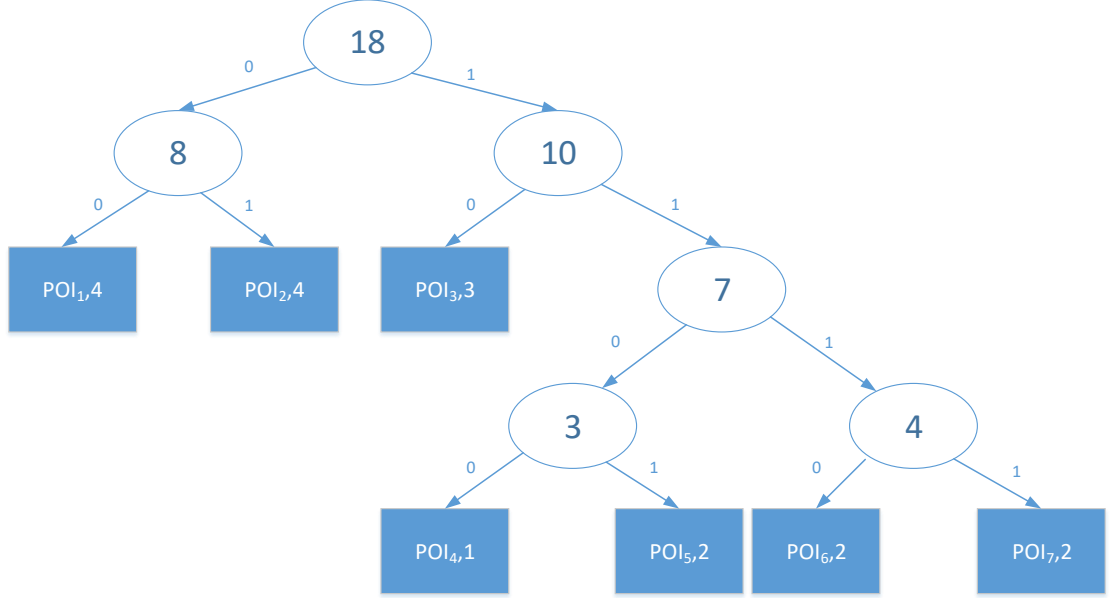


图 3-2 POI 的霍夫曼树表示

霍夫曼树的每个非叶结点都可以看作是一个分类器，来决定该往哪个方向走，叶结点就是 POI。采用层次 softmax 函数计算下一个 POI 的访问概率。表示如下：

$$P(l = l_o | h) = \prod_{j=1}^{|Path(l)|-1} \sigma([m(l, j+1) = ch(m(l, j))]) \cdot v'_{m(l, j)}{}^T h$$

$$[[x]] = \begin{cases} 1 & \text{如果 } x \text{ 为真} \\ -1 & \text{其他情况} \end{cases}, \quad h = \frac{1}{Q} \sum_{q=1}^Q v l_q \quad (3-1)$$

公式中的符号含义如下：

- l_o 表示实际 POI 输出。
- $Path(l)$ 表示从根结点到 POI l 的路径。
- $m(l, j)$ 表示从根结点到 POI l 路径上的第 j 个内结点。
- $ch(m)$ 表示内结点 m 的左孩子。
- $v'_{m(l, j)}$ 是内结点 $m(l, j)$ 的向量表示。
- Q 是签到序列中 POI 的个数。

- vl_q 是 POI l_q 的输入向量。
- $\sigma(\cdot)$ 是 sigmoid 函数。
- h 是在 l_o 之前访问的 POI 的集合。

基于层次 softmax 函数, CBOW 的目标函数如下:

$$\hat{V}'_m, \hat{V}_l = \arg \max_{V'_m, V_l} \prod_i P(l_{o,i} | h_i) \quad (3-2)$$

其中 V'_m 是霍夫曼树中所有内结点的系数集, V_l 是所有 POI 的潜在向量的集合。用户的潜在向量 $V_u = \{v_{u1}, v_{u2}, \dots, v_{u|U|}\}$ 也可以通过类似的方法得到。

3.3 访问概率预测

正如之前所提到的, 用户的签到行为往往受用户偏好和 POI 之间的连续过度影响, 其中用户偏好通过用户的潜在向量表示, POI 之间的连续过度影响可以嵌入到所有 POI 的潜在向量中, 考虑这两个因素, 将用户的潜在向量 V_u 和 POI 的潜在向量 V_l 作为意愿向量组合到一起。如下公式所示, 让 V_ω 作为 LSTM 模型的输入 x_t 。经过 softmax 层后, LSTM 输出用户 u 对于所有 POI 的访问概率。

$$V_\omega = V_l + V_u \quad (3-3)$$

3.4 POI 推荐

根据对图 2-3 的观察, 用户对于 POI 的选择, 很大程度上, 和距离当前的位置有关。也就是说, 模型应该在附近的 POI 上添加更多的权重去匹配用户意图。因此, 在估计出每个 POI 的访问概率之后。根据距离阈值 d 对超过距离的 POI 进行处理。

用户 u 的 POI 推荐列表可以用 R_u 表示如下, 其中 l_c 为用户的当前位置, $(l_q, P(l_q | u, l_c))$ 是 $LSTM(\cdot)$ 的结果, $P(l_q | u, l_c)$ 是用户 u 在时间段 $[t_c, t_c + \tau]$ 对 POI l_q 签到的可能性:

$$R_u = \{(l_q, P(l_q | u, l_c)) | (l_q, P(l_q | u, l_c)) \in LSTM(V_\omega), \wedge distance(l_q, l_c) \leq d\} \quad (3-4)$$

最后, top-N 的 POI 会被推荐给用户。

3.5 参数测定

在 PEU-RNN 模型中，LSTM 是 POI 推荐的基础模型。为了实现更好的性能，每层的单元数和层数，都是构建神经网络的超参数，应事先进行微调。

距离阈值分别在 Gowalla 和 Brightkite 数据集中设置为 1 公里和 5 公里。图 3-3、3-4 显示的是 Gowalla 数据集中层数和每层单元格中的单元数的影响。图 3-5、3-6 显示的是 Brightkite 数据集中层数和每层单元格中的单元数的影响。

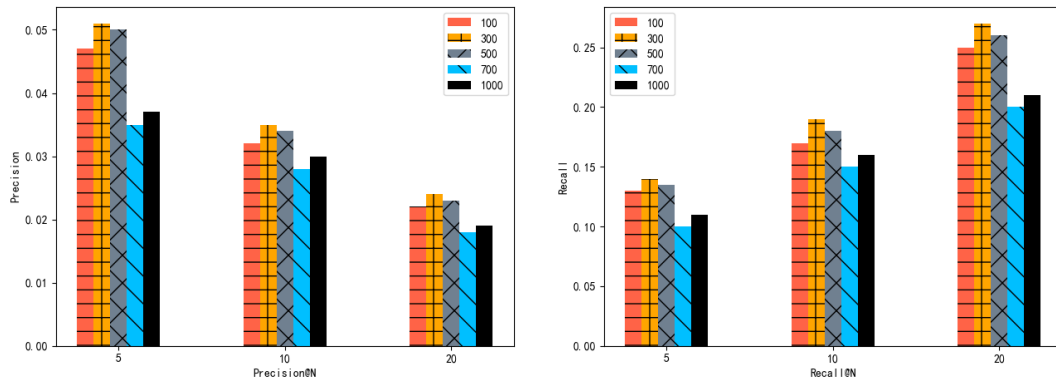


图 3-3 Gowalla 中一层中不同单元数性能对比

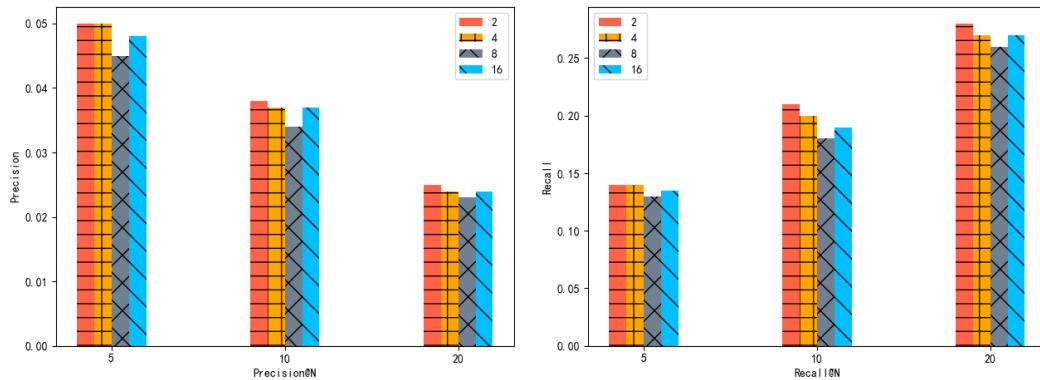


图 3-4 Gowalla 中不同层数性能对比

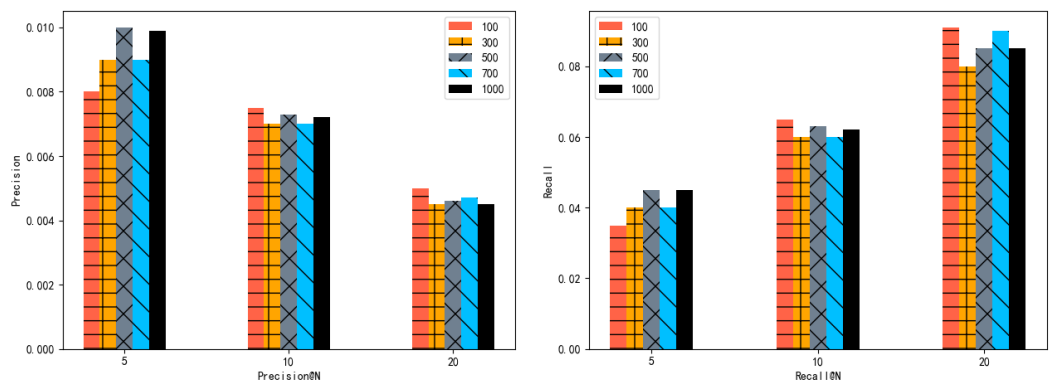


图 3-5 Brightkite 中一层中不同单元数性能对比

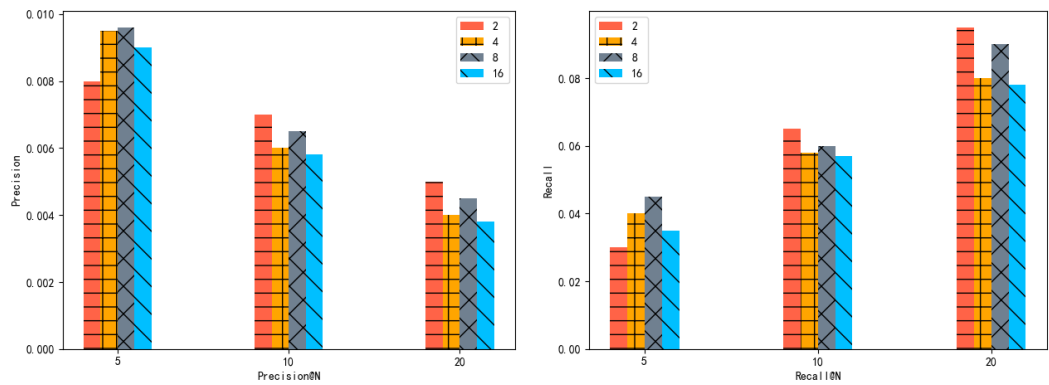


图 3-7 Brightkite 中不同层数性能对比

对于 Gowalla 数据集和 Brightkite 数据集，在一层中最好的单元数是 300 和 100,而 Gowalla 数据集和 Brightkite 数据集的最优层数分别设置为 2,表示如下：

表 3-1 PEU-RNN 最优参数

数据库	每层单元数	层数
Gowalla	300	2
Brightkite	100	2

3.6 实验比较

PEU-RNN 模型采用 CBOW 作为底层的词嵌入技术，从图 2-8、2-11 可以看出连续的过渡影响在 Gowalla 数据集中起着重要的作用，然而，对于图 2-9、2-

12, 在 Brightkite 数据集中, PEU-RNN 模型表现不好。如表 3-2 所示, 对于不同数据集中的签到数量, Gowalla 中要比 Brightkite 中多很多, 对于神经网络而言, 需要大量的数据来训练模型, 因为数据不充分导致 PEU-RNN 模型在性能表现比较差。就 Gowalla 上的实验表明, RNN 能够从用户的签到记录中提取 POI 之间的连续过渡影响。

表 3-2 数据集中的连续签到记录

数据集	3 小时内的连续签到数量	6 小时的连续签到记录
Gowalla	17160 (条)	16791 (条)
Brightkite	6189 (条)	5688 (条)

3.7 本章小结

本章主要是利用神经网络处理 POI 之间的连续过度影响, 通过生成潜在向量来表示 POI 和用户。然后利用 LSTM 模型, 生成 POI 推荐列表。通过实验可以发现, 神经网络在数据集规模较大的情况下, 有着良好的推荐性能。

第 4 章 系统的设计和实现

4.1 需求分析

4.1.1 背景

随着大数据在各行各业的应用,通过分析用户历史行为来提升用户体验的推荐系统,已经在电子商务中扮演着越来越重要的角色,对于 O2O 模式,其隶属于电子商务的范畴,但是基于 POI 的线下特性,传统意义上的推荐算法无法很好的适应 POI 推荐,本文第二章的 UGSE-LR 模型,将用户在 POI 的转移看作是马尔可夫链过程,利用用户的签到数据,发掘 POI 到 POI 之间的连续过度影响。本文第三章的 PEU-RNN 模型,同样是在于发掘 POI 之间的连续过度影响。根据实验结果,PEU-RNN 基于神经网络,需要的大量的数据来训练模型,而 UGSE-LR 模型对于两个数据集都有着良好的推荐性能,对于商圈中的 POI 推荐,本章将基于微信和 UGSE-LR 模型,进行系统的设计和开发,为用户在商圈中不同 POI 之间的转移提供建议,提高用户体验。

4.1.2 可行性研究

当前 O2O 商业模式具有很好的应用前景,已经商用的软件包括美团点评,口碑等,都取得了很好的商用价值,商家在网上发布特价消息或者软件运营部门发布特价活动,都在很大程度上吸引用户进行消费行为。在推荐系统被广泛运用的今天,隶属于电子商务的 O2O 商业模式,如何结合自身特点,利用推荐系统的思想,更好的服务于用户是一个值得被研究的话题,正如本文第二章所提出的思想。利用用户的历史签到数据,和当前位置,可以将用户偏好,区域影响,还有连续过度影响这三个影响因子作为推荐 POI 的计分凭据。然后将得分最高的 POI 推荐给用户。

4.1.3 功能需求

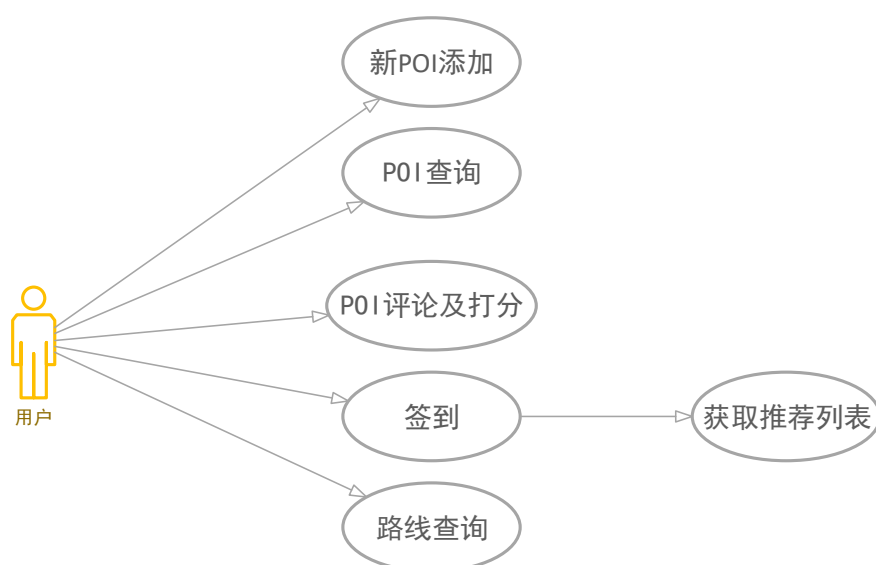


图 4-1 用户用例图

新 POI 添加：对于系统中没有的 POI，用户可以自主创建，创建信息包括地理位置、名称、描述、POI 类型以及创建理由。

POI 查询：用户可以设置查询条件，查询条件包括区域位置、POI 关键字、评分等。查询信息包括 POI 位置信息，用户评分信息和评论信息。

POI 评分：用户完成签到行为后，可以就 POI 环境以及服务态度等进行评论和评分，评分为五分制。

签到：用户到达目标 POI，进行签到行为，系统根据用户历史记录和当前位置生成推荐列表。

路线查询：用户选择推荐列表中的推荐 POI，或者自行搜寻 POI 时，系统会为用户推荐到达路线。

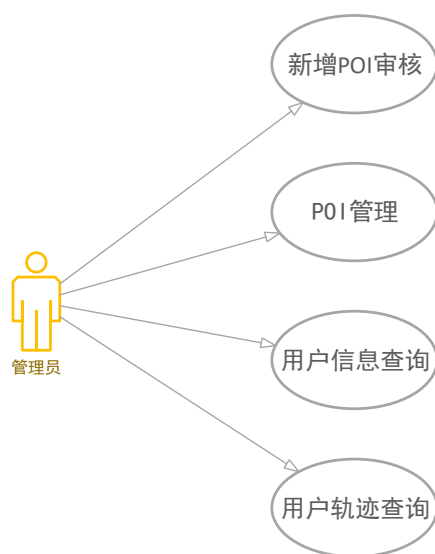


图 4-2 管理员用例图

新增 POI 审核：对于用户添加的新 POI 进行审核工作，包括基本信息验证和 POI 合法性验证。

POI 管理：管理员可以查询 POI 的详细信息，修改、删除 POI 数据，审核用户新添加的 POI 信息。

用户信息查询：可以对用户的基本信息进行查询，查询内容包括身份信息、账号详情等。

用户轨迹查询：管理员可以跟踪用户轨迹，查询用户的所有签到数据并进行图形化展示。

4.2 系统设计

4.2.1 系统结构设计

商圈中连续 POI 推荐系统主要是收集用户的签到信息，根据用户的签到行为，为用户生成 POI 的推荐列表，提供在商圈中 POI 的转移建议，从而提升用户体验。根据需求分析文档，设计了用户管理、POI 管理、用户签到和 POI 查询模块。系统的功能结构图如图 4-3 所示：

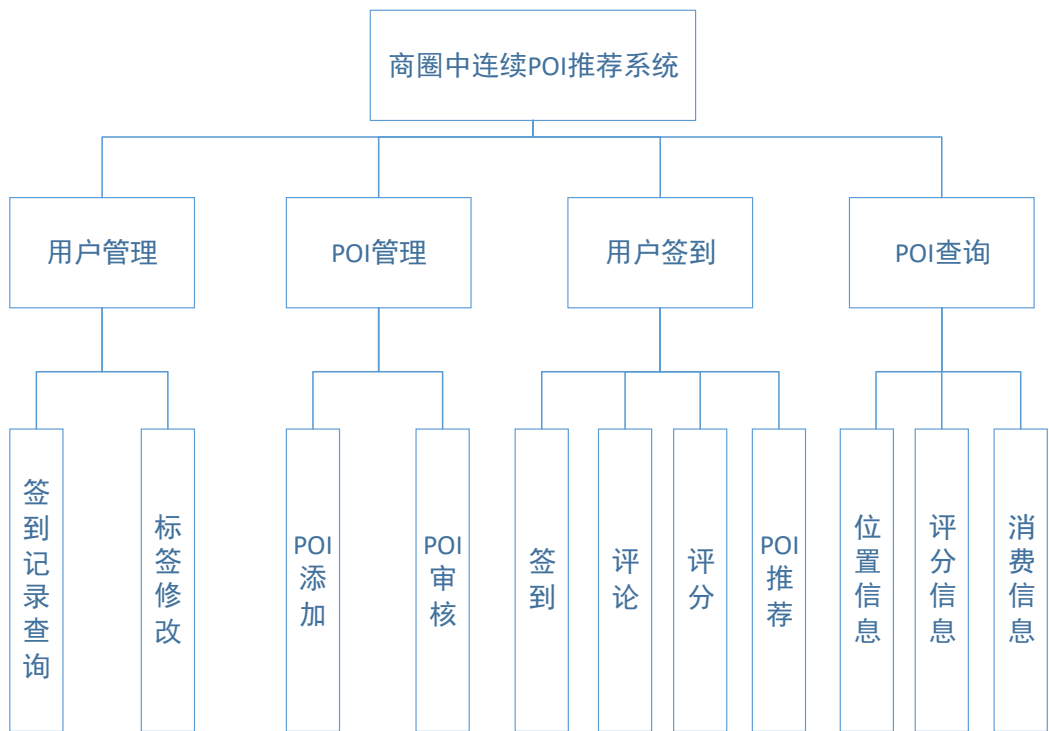


图 4-3 功能结构图

如图 4-4 所示，商圈中连续 POI 推荐系统包括客户端和管理端。客户端服务于用户，对于用户而言，可以在客户端完成对 POI 的查询和签到行为。系统可以根据用户的签到行为，为用户生成 POI 推荐列表。管理端可以帮助管理员完成 POI 的管理，包括 POI 的添加和审核工作。

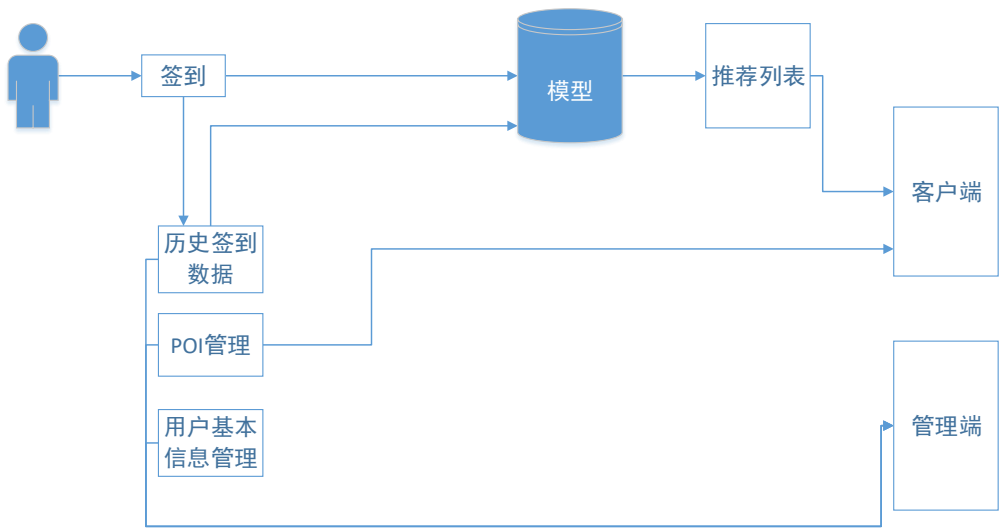


图 4-4 系统架构图

4.2.2 系统初始化

4.2.2.1 冷启动问题

当新用户进入系统，或者新 POI 添加进系统时，系统存在冷启动问题，分别为系统冷启动、用户冷启动以及 POI 冷启动。

系统冷启动：系统最开始状态，因为没有用户的签到记录，而采用非个性化推荐。根据用户的签到位置，就第二章提出的网格思想，找出所有的临近网格中的 POI，然后按照 POI 的评分和用户标签，完成推荐列表。

用户冷启动问题：新用户在没有历史签到数据的情况下，无法根据协同过滤计算出用户偏好，只能根据用户的当前签到位置，计算出区域影响和 POI 之间的连续过度影响，这里的用户偏好可以根据用户的注册信息建立用户特征模型来获得。同性别，同年龄段，同标签用户，往往拥有相同的兴趣爱好。

POI 冷启动：对于新添加的 POI，用户的历史签到数据往往无法覆盖，这里根据 POI 标签与用户标签，直接添加到用户的推荐列表中，对于每个用户而言，所有的新 POI 都有且只有一次冷启动机会。

对于冷启动问题，我们引入标签体系，为每个用户添加个性化标签，同样的，为每个 POI 添加标签。

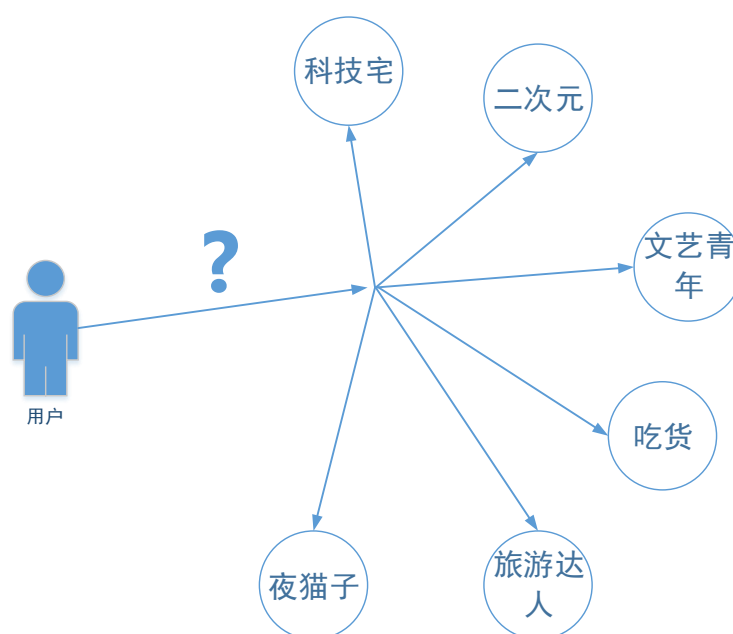


图 4-5 用户标签图

用户注册时,可以为自己贴上标签,包括科技宅、二次元、文艺青年、吃货、旅游达人、夜猫子等。方便系统为用户建模模型。在无历史签到数据的情况下,为用户实现推荐。

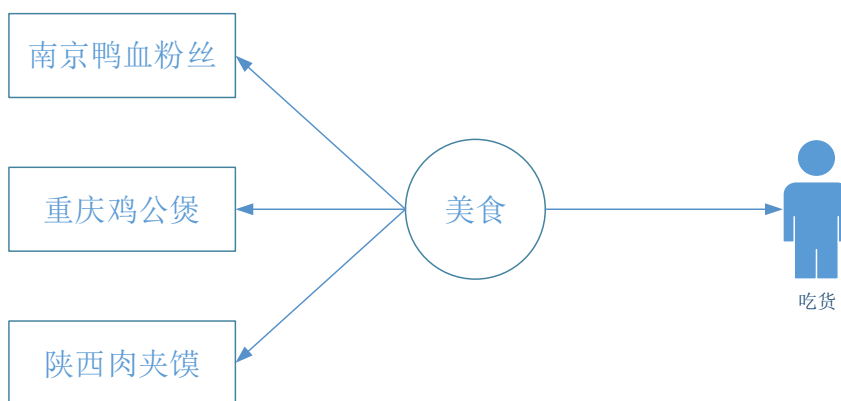


图 4-6 吃货标签映射关系

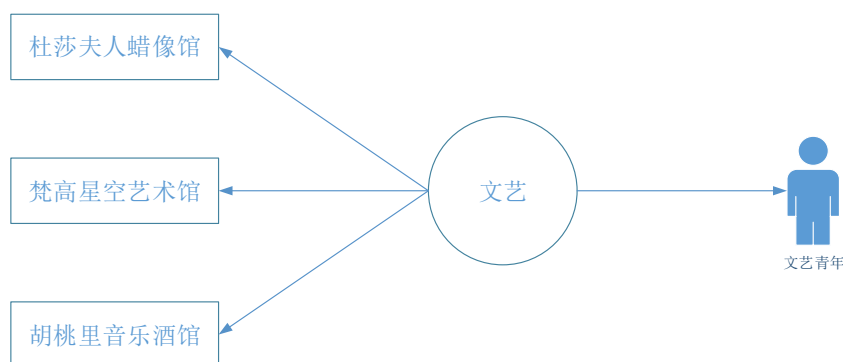


图 4-7 文艺青年标签映射关系

POI 标签：用户在创建新 POI 时，会添加 POI 类型。例如武汉街道口新开了一家鸭血粉丝店,对于系统而言,这里是一个新 POI,用户无法进行签到行为,此时用户可以选择新 POI 的添加,这里除了对 POI 进行简单描述以外,也要为 POI 贴上标签。

4.2.2.2 用户偏好的获取

用户的签到记录无法表示自身对于 POI 的偏好信息,也就是说,不能简单的根据用户的签到与否来得出用户对于 POI 的喜好问题,这里,我们对模型进行改进,引入评分信息,使得用户可以对于自己的签到位置发布自己的态度。

4.2.3 数据库设计

数据库的实体-联系如图 4-8 所示：

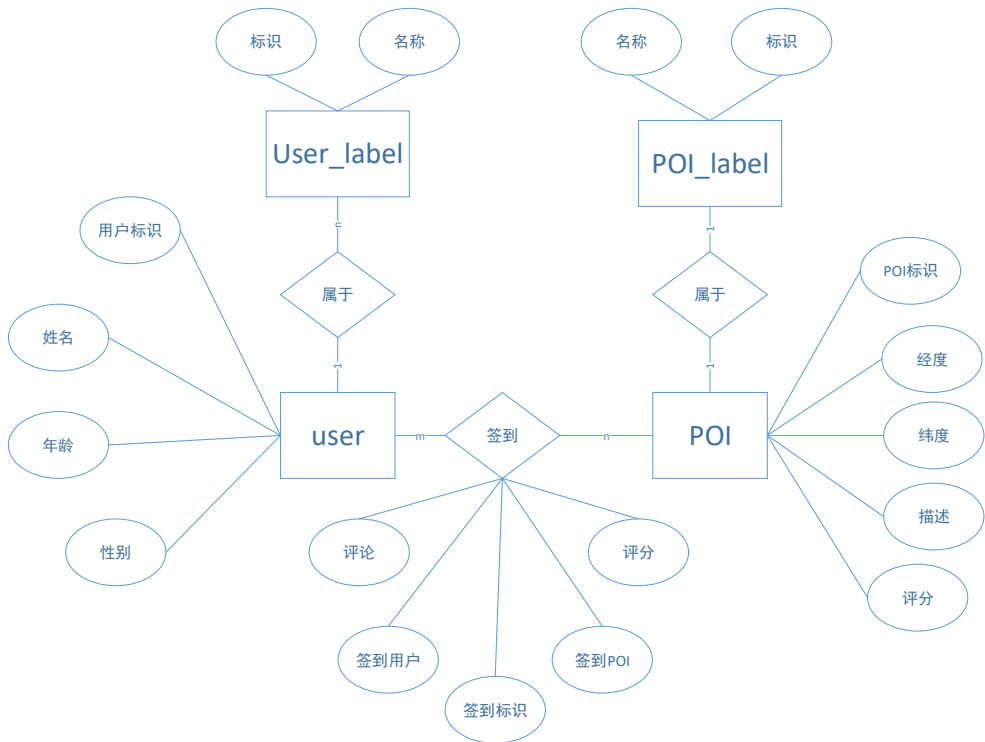


图 4-8 E-R 图

数据库共包含五张表，分别是用户信息表，POI 信息表，用户签到表，POI 标签表 and 用户标签表。

1) 用户信息表 (user_info)

主要存储注册用户的基本信息，包括账号、密码、姓名、性别、年龄和用户标签。

表 4-1 用户信息表

属性名	字段名	数据类型	是否可为空	备注
账号	id	char(10)	否	主键
密码	password	varChar(20)	否	
姓名	name	char(10)	是	
性别	sex	char(4)	是	
年龄	old	int	是	
用户标签	user_label	char(10)	否	外键

2) POI 信息表 (POI_info)

主要存储 POI 的基本信息, 包括 POI 标识、POI 经纬度、POI 描述、POI 标签、POI 评分等。

表 4-2 POI 信息表

属性名	字段名	数据类型	是否可为空	备注
POI 标识	POI_id	char(10)	否	主键
POI 经度	longitude	varChar(20)	是	
POI 纬度	latitude	varChar(20)	是	
POI 描述	description	varChar(200)	是	
POI 评分	score	int	是	
POI 标签	POI_label	varChar(20)	否	外键

3) 用户签到表 (user_checkins)

主要用于存储用户的签到记录, 包括签到标识、签到用户 id、用户的签到时间、签到 POI 的 id、用户的签到描述和签到评分。

表 4-3 用户签到表

属性名	字段名	数据类型	是否可为空	备注
签到标识	checkins_id	char(10)	否	主键
签到时间	checkins_time	datetime	是	
签到用户	checkins_user	char(10)	否	外键
签到 POI	checkins_POI	char(10)	否	外键
签到评论	comment	varChar(200)	是	
签到评分	grade	int	是	

4) POI 标签表 (POI_label)

主要是存储 POI 标签的基本信息, 包括 POI 标签标识和 POI 标签名称。

表 4-4 POI 标签表

属性名	字段名	数据类型	是否可为空	备注
POI 标签标识	POI_label_id	char(10)	否	主键
POI 标签名称	label_name	char(10)	是	

5) 用户标签表 (user_label)

主要是存储用户标签的基本信息，包括用户标签标识和用户标签名称。

表 4-5 用户标签表

属性名	字段名	数据类型	是否可为空	备注
用户标签标识	user_label_id	char(10)	否	主键
用户标签名称	label_name	char(10)	是	

4.3 主要界面

系统主要包括四个界面，分别是首页、POI 详情页、用户详情页、签到页和推荐列表页。

1) 首页

用户可以通过首页选择商圈，获取商圈内的 POI 列表，也可通过搜索条件，筛选相应的 POI 类型。用户根据自己所在位置，进行签到行为。



图 4-9 系统首页

2) POI 详情页

用户可以通过点击查看详情按钮，查看 POI 的详细信息，包括 POI 的位置信息、联系方式、营业时间、评分以及用户评论等，并可完成签到行为。



图 4-9 POI 详情页

3) 用户详情页

用户可以通过点击头像，查看自己的详细信息，包括性别、年龄、所在城市、以及自己的标签类别。

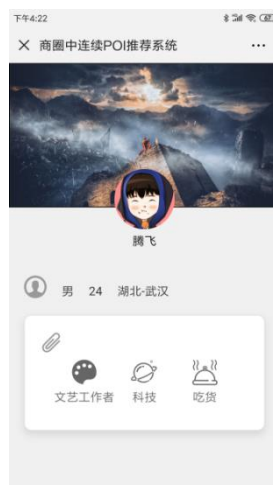


图 4-10 用户详情页

4) 签到详情页

用户可以选择 POI 进行签到行为，签到内容包括评分信息和评论信息。另外，用户可以选择是否匿名评价来发布自己的签到信息。



图 4-11 签到页

5) 获取推荐列表

用户在完成签到行为后，系统会根据 UGSE-LR 模型，为用户生成 POI 推荐列表。推荐列表展示 POI 的基本信息，包括 POI 名称、评分、以及位置信息。用户可以通过点击查看详情，获取更多 POI 的信息。



图 4-12 推荐列表页

4.4 本章小结

本章主要是根据第二章的模型进行了系统的设计和实现，根据连续 POI 推

荐的思想，完成商圈中连续 POI 推荐系统的设计和实现。并引入标签体系解决系统的冷启动问题。

第 5 章 总结与展望

5.1 工作总结

O2O 商业模式的火爆，让商圈中的 POI 推荐成为热门课题，对于商圈中 POI 而言，其往往具有很强的区域聚集性。用户在选择签到地点的时候，容易被时间上下文、空间上下文、用户社会关系、POI 主题等所影响，用户在 POI 之间的转移，往往也具有连续性，这些特性决定了 POI 推荐的复杂性。本文就如何有效的利用用户的历史签到数据，提升用户在商圈中的体验，进行了如下工作：

1) 调研了 O2O 商业模式的发展现状，对于如何提升用户在商圈中的体验，提出了 POI 推荐问题。结合传统推荐算法的特性，分析并总结了 POI 在推荐中需要考量的特性问题。

2) 针对 POI 推荐，调研了现阶段的 POI 推荐问题，并结合当前最为流行的连续推荐，将 POI 推荐划分为实时推荐和非实时推荐。结合商圈中的 POI 特性，提出采用实时推荐来解决用户在商圈 POI 之间的推荐问题

3) 对于连续推荐问题，本文提出了两个模型来处理商圈中的 POI 推荐问题，并结合 Gowalla 和 Brightkite 两个数据集，设计了相关实验。根据准确率和召回率，分析了不同模型的适用场景。

4) 根据实验结果，采用 UGSE-LR 模型作为商圈中连续 POI 推荐系统的模型。并针对系统的冷启动问题，提出了标签体系。设计并实现了系统。

5.2 展望

本文针对商圈中的 POI 推荐问题，提出了连续推荐的处理思路。但是在实际的应用中，仍有若干问题需要解决：

1) 用户偏好的确定问题：系统对于用户偏好，引入了评分体系，虽然评分可以在很大程度上表示用户对于 POI 的偏好问题，但是，用户的评论信息并没有很好的被利用上，这里可以加入细粒度情感分析，来发掘用户的偏好详情

问题。

2) 对于模型 UGSE-LR 而言, 区域影响因子的确定不能简单的用网格划分, 根据实际情况, 我们可以将一个商圈作为一个区域。那么, POI 的经纬度显然不能作为划分的依据, 需要人员实地调研, 进行商圈的划分。

致谢

两年的研究生生涯，马上就要结束了，在漫漫的求学生涯的最后阶段，有太多需要回味和反思的。这里，我首先要感谢我的导师，刘永坚老师，作为求学生涯的最后一任老师，刘老师以他的大格局深刻影响着我。从计算机出发，以学科融合的态度对待出版行业，在出版行业实现计算机的新型应用，实现出版融合。创造引领行业的新产品。这都是本科阶段，我不曾有过的想法。然后，我也要感谢我的学术指导人，解庆老师，在论文的撰写过程中，我遇到了太多的问题，解老师以自己丰富的经验和专业的学术功底，给我提了很多建设性建议，无论是从论文的选题开始，还是相关研究的调研，以及论文的撰写，解老师都给了我很多的帮助。这里，我还要感谢我的同学们，学术的道路注定是孤独的，在探索真知的过程中，我们都曾迷惘过，但路途中，有了你们的陪伴，再孤独的旅程也会是精彩飞扬。最后，要感谢我的父母。求学二十载，太多的心血放在了我的身上，深夜苦读的台灯旁，永远是你们的陪伴，无论是生活中，还是学习上，你们都付出了太多，初高中的陪伴，到后来我独自踏上大学，再到如今的研究生生涯。你们一直是我继续下去的理由，感谢有你们。

参考文献

- [1] 商务部电子商务和信息化司. 中国电子商务报告 2017[R]. 北京: 中国商务出版社, 2018.
- [2] Linden G, Smith B, York J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering[J]. IEEE Internet Computing, 2003, 7(1): 76-80.
- [3] Shardanand U. Social Information Filtering: Algorithm for Automating 'Word of Mouth' [J]. Proc Chi, 1995, 110(1): 210-217.
- [4] 李国徽, 王洪亚. 移动计算系统中位置数据库组织[J]. 计算机工程与科学, 2003, 25(2): 76-78.
- [5] Fano A E. Shopper's eye: using location-based filtering for a shopping agent in the physical world[C]. International Conference on Autonomous Agents. ACM, 1998: 416-421.
- [6] Espinoza F, Persson P, Sandin A, et al. GeoNotes: Social and Navigational Aspects of Location-Based Information Systems[C]. International Conference on Ubiquitous Computing. Springer-Verlag, 2001: 2-17.
- [7] Bao J, Zheng Y, Wilkie D, et al. Recommendations in location-based social networks: a survey[J]. Geoinformatica, 2015, 19(3): 525-565.
- [8] Zheng Y, Chen Y, Xie X, et al. GeoLife2.0: A Location-Based Social Networking Service[C]. Tenth International Conference on Mobile Data Management: Systems, Services and MIDDLEWARE. IEEE, 2009: 357-358.
- [9] 吴芝新. 简析 O2O 电子商务模式[J]. 重庆科技学院学报(社会科学版), 2012(13): 73-74.
- [10] 孙悦, 郭醒, 徐欣欣. O2O 电子商务模式剖析[J]. 电子商务, 2013(11): 5-5.
- [11] 姜珊. 基于位置的社会网络中多因素感知 POI 推荐策略[D]. 江西财经大学, 2017.
- [12] Rendle S. Factorization Machines[C]. international conference on data mining, 2010: 995-1000.
- [13] Wang W, Yin H, Chen L, et al. Geo-SAGE: A Geographical Sparse Additive Generative Model for Spatial Item Recommendation[C]. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015: 1255-1264.

- [14] Yuan Q, Cong G, Sun A, et al. Graph-based Point-of-interest Recommendation with Geographical and Temporal Influences[C]. conference on information and knowledge management, 2014: 659-668.
- [15] Zhang J, Chow C, Li Y, et al. LORE: exploiting sequential influence for location recommendations[C]. advances in geographic information systems, 2014: 103-112.
- [16] 任星怡, 宋美娜, 宋俊德. 基于用户签到行为的兴趣点推荐[J]. 计算机学报, 2017(1): 28-51.
- [17] Lu Y S, Shih W Y, Gau H Y, et al. On successive point-of-interest recommendation[J]. World Wide Web-internet & Web Information Systems, 2018: 1-23.
- [18] Zheng Y, Zhang L, Xie X, et al. Mining interesting locations and travel sequences from GPS trajectories[C]. International Conference on World Wide Web. ACM, 2009: 791-800.
- [19] Song L, Kotz D, Jain R, et al. Evaluating location predictors with extensive Wi-Fi mobility data[C]. Joint Conference of the IEEE Computer and Communications Societies. IEEE, 2004: 1414-1424 vol.2.
- [20] Cheng H, Ye J, Zhu Z, et al. What's Your Next Move: User Activity Prediction in Location-based Social Networks[C]. siam international conference on data mining, 2013: 171-179.
- [21] Ying J J, Lu E H, Kuo W N, et al. Urban point-of-interest recommendation by mining user check-in behaviors[C]. knowledge discovery and data mining, 2012: 63-70.
- [22] Yin H, Zhou X, Shao Y, et al. Joint Modeling of User Check-in Behaviors for Point-of-Interest Recommendation[C]. conference on information and knowledge management, 2015: 1631-1640.
- [23] Yuan Q, Cong G, Ma Z, et al. Time-aware point-of-interest recommendation[C]. international acm sigir conference on research and development in information retrieval, 2013: 363-372.
- [24] Yao L, Sheng Q Z, Qin Y, et al. Context-aware Point-of-Interest Recommendation Using Tensor Factorization with Social Regularization[C]. international acm sigir conference on research and development in information retrieval, 2015: 1007-1010.
- [25] Ying Y, Chen L, Chen G, et al. A temporal-aware POI recommendation system using context-aware tensor decomposition and weighted HITS[J]. Neurocomputing, 2017: 195-205.
- [26] Ye M, Yin P F, Lee W, et al. Exploiting geographical influence for collaborative point-of-interest recommendation[C]. international acm sigir conference on research and development in information retrieval, 2011: 325-334.

- [27] Liu B, Fu Y, Yao Z, et al. Learning geographical preferences for point-of-interest recommendation[C]. knowledge discovery and data mining, 2013: 1043-1051.
- [28] Zhao X, Ma Z, Zhang Z. A novel recommendation system in location-based social networks using distributed ELM[J]. Memetic Computing, 2017(3): 1-11.
- [29] Gao H, Tang J, Liu H, et al. gSCorr: modeling geo-social correlations for new check-ins on location-based social networks[C]. conference on information and knowledge management, 2012: 1582-1586.
- [30] Ference G, Ye M, Lee W, et al. Location recommendation for out-of-town users in location-based social networks[C]. conference on information and knowledge management, 2013: 721-726.
- [31] Yang D, Zhang D, Yu Z, et al. A sentiment-enhanced personalized location recommendation system[C]. acm conference on hypertext, 2013: 119-128.
- [32] Liu B, Xiong H. Point-of-Interest Recommendation in Location Based Social Networks with Topic and Location Awareness.[C]. siam international conference on data mining, 2013: 396-404.
- [33] Cheng C, Yang H, Lyu M R, et al. Where you like to go next: successive point-of-interest recommendation[C]. international joint conference on artificial intelligence, 2013: 2605-2611.
- [34] He J, Li X, Liao L, et al. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns[C]. national conference on artificial intelligence, 2016: 137-143.
- [35] Zhao S, Zhao T, Yang H, et al. STELLAR: spatial-temporal latent ranking for successive point-of-interest recommendation[C]. national conference on artificial intelligence, 2016: 315-321.
- [36] He J, Li X, Liao L, et al. Inferring a personalized next point-of-interest recommendation model with latent behavior patterns[C]. national conference on artificial intelligence, 2016: 137-143.
- [37] Zhao S, Lyu M R, King I. STELLAR: Spatial-Temporal Latent Ranking Model for Successive POI Recommendation[M]. Point-of-Interest Recommendation in Location-Based Social Networks. 2018.
- [38] Feng S, Li X, Zeng Y, et al. Personalized ranking metric embedding for next new POI recommendation[C]. international conference on artificial intelligence, 2015: 2069-2075.
- [39] Liu X, Liu Y, Li X, et al. Exploring the context of locations for personalized location recommendations[C]. international joint conference on artificial intelligence, 2016: 1188-1194.

- [40] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositionality[J]. neural information processing systems, 2013: 3111-3119.
- [41] Weston J, Bengio S, Usunier N, et al. Large scale image annotation: learning to rank with joint word-image embeddings[J]. european conference on machine learning, 2010, 81(1): 21-35.
- [42] Feng S, Cong G, An B, et al. POI2Vec: Geographical Latent Representation for Predicting Future Visitors[C]. national conference on artificial intelligence, 2017: 102-108.
- [43] Xie W, Bindel D, Demers A J, et al. Edge-Weighted Personalized PageRank: Breaking A Decade-Old Performance Barrier[C]. knowledge discovery and data mining, 2015: 1325-1334.
- [44] Sehgal U, Kaur K, Kumar P, et al. The Anatomy of a Large-Scale Hyper Textual Web Search Engine[C]. international conference on computer and electrical engineering, 2009: 491-495.
- [45] Balmin A, Hristidis V, Papakonstantinou Y, et al. Objectrank: authority-based keyword search in databases[J]. very large data bases, 2004: 564-575.
- [46] Backstrom L, Leskovec J. Supervised random walks: predicting and recommending links in social networks[J]. web search and data mining, 2011: 635-644.
- [47] Feng W, Wang J. Incorporating heterogeneous information for personalized tag recommendation in social tagging systems[C]. knowledge discovery and data mining, 2012: 1276-1284.
- [48] Gleich D F. PageRank Beyond the Web[J]. Siam Review, 2015, 57(3): 321-363.
- [49] 朱凡微, 吴明晖, 应晶. 高效个性化 PageRank 算法综述[J]. 中国科技论文, 2012, 07(1): 7-13.
- [50] Fogaras D, Racz B. Towards Scaling Fully Personalized PageRank[J]. workshop on algorithms and models for the web graph, 2004: 105-117.
- [51] Rendle S, Freudenthaler C, Schmidtthieme L, et al. Factorizing personalized Markov chains for next-basket recommendation[J]. the web conference, 2010: 811-820.