论 文

基于 PMR 架构的兴趣点推荐研究

王峰, 余伟*, 李石君

武汉大学计算机学院, 武汉 430072

* 通信作者. E-mail: afengcom@163.com

收稿日期: 2014-12-02; 接受日期: 2015-01-08; 网络出版日期: 2015-10-09

国家自然科学基金 (批准号: 61272109) 和武汉大学博士研究生自主科研项目 (批准号: 2012211020211) 资助

摘要 基于位置的移动社交网络, 如微博、微信等为移动社交网络的研究提供了大量用户历史行为信息, 诸如用户资料、签到时间、位置轨迹、微博内容等. 它们为基于情境的用户实时兴趣点推荐的研究提供了契机. 本文通过借鉴 PMJ 模型的思想, 建立了一种全新的位置推荐系统原型 PMR 架构, 它最大的优势在于, 通过把最新的情境感知技术和计算方法与 PMJ 架构的优势相结合, 能够对用户反馈信息实时处理后依据该信息对推荐方案进行分值和排序更新, 因此这种以用户需求为核心的推荐方案更加契合用户需求. 在此架构的基础上, 提出了一整套用于情境感知、存储、推荐和反馈的计算方法, 较好地解决了情境计算研究领域基于用户反馈机制的自适应个性化推荐这一关键问题. 实验结果表明, 该系统架构具有较高的召回率 (96.93%) 和准确率 (87.27%), 以及较好的用户满意度 (80.12%).

关键词 基于位置的社交网络 PMR 架构 兴趣点推荐 情境感知 城市计算

1 引言

在诸如微博、微信等社交网络中, 用户能通过移动服务终端共享信息, 如用户签到 check-ins, 用户共享的 GPS 位置和时间运动轨迹等. 这些信息显示了用户曾经到过哪里, 可能和其他人同时去过某地, 出入某活动区域的频次和日常活动的范围等. 通过对上述信息进行挖掘, 可以形成一套基于用户历史行为特征的兴趣库. 当用户随机出现于某地时, 根据兴趣库可以为用户推荐附近他感兴趣的位置点和社交场所. 同时, 该用户还可以把这些推荐信息共享给其他用户. 推荐信息的共享有助于分析其他用户是否具有和该用户类似的兴趣. 位置推荐架构中最受推崇的方法是协同过滤的位置推荐技术. 该方法优势在于, 采用动态聚类算法能够有效地把具有特征相似用户、相似活动和相似位置等聚集起来, 并且当出现新轨迹数据时, 它可以把新数据归类到旧数据中. 但该方法缺少对用户反馈信息的处理和及时更新, 因而显得不太人性化. 本文通过获取和处理用户反馈信息, 并借鉴 PMJ(perception memory judgement) 架构的思想, 提出了一种基于用户反馈的全新位置推荐系统原型 PMR(perception memory recommendation) 架构. 因此, 文章的主要贡献和创新点可概括为如下几点.

(1) 由于情境感知的思想和技术在移动互联网领域中本身就是目前该研究领域的最新方法, 因而从解决问题的思路上来讲, 它为 PMR 架构的建立和对现存方法性能上的改进提供了科学的理论依据.

引用格式: 王峰, 余伟, 李石君. 基于 PMR 架构的兴趣点推荐研究. 中国科学: 信息科学, 2015, 45: 1503–1520, doi: 10.1360/N112014-00251

(2) PMR 架构给出了解决情境感知、存储、推荐的新方法, 与最先进的协同过滤推荐方法相比, 推荐的用户满意度有了很大程度的提升.

(3) PMR 架构中对于情境推荐的 3 种推荐方法: 快速推荐、综合推荐和快速综合推荐. 它们都具有较高的召回率、推荐准确率和推荐用户满意度.

(4) PMR 架构通过用户对推荐的反馈机制能够对推荐方案进行打分并排序, 以用户的个性化需求为主要依据, 尽量把用户认为满意的推荐方案推荐给用户, 这样使推荐更人性化, 更贴近用户的真实需求.

(5) 与传统的机器学习 (ML) 模型相比, PMR 架构的优势在于, 它不但能够基于已存在的知识库进行学习, 而且当新用户的个性化需求在知识库中不存在时, 该架构通过实时的感知 P 过程提取每个用户新的情境信息, 并通过类 User-User, User-Item 等协同聚类技术将具有相似特征的用户和事物的属性相结合用于辅助计算方案的得分, 然后进行分值的 TOP 排序, 同样可以为用户的个性化需求进行精确推荐. 而感知 P 过程的实时性则体现在, 先进的情境感知技术能够对周边环境信息做到实时采集和提取.

2 相关工作

(1) 基于协同过滤技术的推荐系统. 协同过滤技术 (collaborative filtering, CF) 提供了个性化推荐^[1,2], 这种技术在实践中非常受欢迎^[3~6]. CF 假设过去有相似行为的用户会喜欢同样的物品 (基于用户的协同过滤), 并且如果用户过去曾经青睐的物品, 今后有类似物品出现时, 他同样会喜欢 (基于物品的协同过滤). 此外, 还有一种基于模型的推荐系统^[7~9]. 虽然上述方法能够改进后应用于基于位置的社交网络 (LBSNs) 中, 但其算法并不能表现出较好性能, 其原因是这类方法没有恰当考虑社会 and 空间信息.

(2) 兴趣点推荐. Ye 等^[10] 基于用户协同过滤框架下的社会影响力和基于模型 (一种贝叶斯 (Bayes) 协同过滤算法) 的空间影响力建模. 为了利用社会影响力, 研究利用用户的朋友作为推荐的桥梁; 而为了利用空间影响力, 研究假设用户访问兴趣点的概率由他们间的距离决定. 该方法虽然同时考虑了社会 and 空间信息, 但却没有考虑用户对这些兴趣点推荐的反馈信息.

(3) 基于社交网络的推荐. Pham 等^[11] 研究发现, 社交网络中用户和物品聚类与采用这些信息可以增强协同过滤技术. Yildirim 和 Krishnamoorthy^[12] 建立了一种物品图, 图中的两两物品间被链接起来并赋予权重, 权重用于表示两物品间的相似程度. 为保证物品图连贯性, 即使是几乎不相似的物品也会被赋予很小的权重. 因而导致了该方法不太适合大数据集合. Konstantas 等^[13] 研究一种音乐社交网络协同过滤方法, 该方法可用于预测音乐的点击数量. 他们创建了一种包含用户、音乐和标签的异构图, 并且一种关于图的随机游走算法优于传统的协同过滤方法. 然而这种方法并不适合解决基于位置的社交网络中的推荐问题.

(4) 基于位置的服务与推荐. Scellato 等^[14] 研究采用用户签到记录预测关系链接问题. 他们发现, 基于位置的服务中只需要靠关注朋友间的关系和用户朋友所在的位置就可极大地有利于关系链接的预测. Bao 等^[15] 提出了一种基于 GPS 轨迹的位置推荐框架. 一组 GPS 轨迹是一系列在一定时间内收集到的带时间戳的经纬度对. 停留点 (用户花足够长时间停留的地理区域范围) 能够从 GPS 日志中识别出来并当作位置点对待. 和用户签到数据相比, GPS 数据更加密集, 并且包含很多日常活动信息. 基于这类 GPS 数据特征, 利用用户和其停留点的协同聚类能加强协同过滤推荐. Levandoski 等^[16] 研究带位置分类层次的位置推荐 (例如, 食物、商店等分类及其子类). 他们的主要发现是不同用户对于

不同类型的位置有各自独到的见解,因此在推荐中他们应该被区别对待. Fu 等^[17]考虑采用位置信息进行物品推荐. 在他们的研究中,一件物品因空间和非空间评分的差异而具有其空间属性. 每个位置层次被建立以辅助推荐过程. 这类方法的研究侧重也同样没考虑用户对推荐内容的反馈,而这些反馈才是表达用户真正需求的信息. 本文通过参考基于人类认知基本原理的感知、记忆和决策 (PMJ) 模型^[18]和智能的感知、记忆、预测 (PMP) 框架^[19],引入情境感知技术,对用户历史行为进行挖掘,建立了一种全新的位置推荐系统原型 PMR 架构,它通过把最新的情境感知技术和计算方法与 PMJ 架构的优势相结合,能够对用户反馈信息进行实时的处理和对推荐方案的及时更新,通过指定用户的反馈计算推荐方案针对用户的个性化得分,使推荐方案能够针对用户的兴趣自适应修正,以此更加满足用户需求.

3 PMR 系统架构

所谓 PMR 架构,就是本文兴趣点推荐的 3 个核心阶段:感知 P(perception)、存储 M(memory) 和推荐 R(recommend). 其中,感知是对人脑认知现实世界行为的模拟,内部系统感知单元 P 针对外界刺激进行认知,把不同刺激通过各种传导通路传递到感知单元后,对各种刺激进行分析并按类编码,然后把各种分类编码传递给记忆单元,到达记忆阶段;记忆阶段是对人脑存储事物的一种模拟,与人脑存储模式类似,分为对时间和空间两种模式的记忆^[20,21],它们分别被称之为时间模式记忆和空间模式记忆,定义在 3.1 小节给出.

图 1 是对用户历史行为从感知到推荐的流程描述. 首先,PMR 系统架构接收外界各种刺激,这些刺激经由通路进入感知阶段被感知模块感知. 信息被感知后由信息通路传递给存储模块进入存储阶段 (图中 ④). 存储单元根据信息的重要性也分为长时和短时存储两种方式. 经感知阶段分析得出的相关情境均被存储于此,高相关情境会长时存储,而低相关情境会短时存储甚至被挤出存储队列而消失. 第 3 个重要模块是推荐模块,该模块用于情境推荐,推荐的理论依据来源于存储到推荐的模块映射过程 (图中 ⑤),然后做出推荐行为对情境进行推荐. 反馈流程分 3 个过程:推荐对存储的反馈 (图中 ⑥)、存储对感知的反馈 (图中 ⑦) 和推荐对感知的反馈 (图中 ⑧). 通过反馈过程来实时更新 3 个核心模块的状态 (图中 ①~③),借此提高系统架构的推荐准确率. 根据时态把情境划分为 3 段:过去情境 SP,当前情境 SN,将来情境 SF. 那么,当用户处于当前情境并感知到情境 SN 时,架构会基于 SP 到 SN 这一时间段的信息,直接给出推荐结果 (图中 ⑧). 由于 SN 是 SF 的过去情境,因此推荐过程与上述类似.

3.1 定义

“情境”感知依赖于“场景”,该技术源于对普适计算的研究,最早由 Schilit 于 1994 年提出.

定义 1 (情境感知 (context-awareness)) 通过计算机认知系统把现实中的情境映射到计算机模拟出的智能系统中,使其能感知现实世界的变化. 其实现过程在 3.2 小节感知实现中给出.

定义 2 (情境存储 (context-memory)) 把感知情境经过分析,分类编码存储到计算机的存储单元内就是情境存储. 根据存储模式的差异,可分为对时间的存储 (时间模式记忆) 和对空间的存储 (空间模式记忆),如定义 3 记忆模式.

定义 3 (记忆模式) 计算机对现实世界中的情境进行映射和存储的方式. 情境被映射到计算机的记忆单元中对应两大类情境记忆,即对时间的记忆 (时间模式记忆) 和对空间的记忆 (空间模式记忆). 时间模式记忆在计算机中对应于时间序列编码,而空间模式记忆在计算机中对应于空间划分元素

图 1 (网络版彩图)PMR 系统架构
Figure 1 (Color online)PMR framework

编码, 实现过程在 3.3 小节的存储实现中给出.

定义 4 (兴趣点 (points-interest)) 描述用户 u 在时间 t 出现在位置 l 的行为轨迹, 即用户感兴趣的位置点信息.

定义 5 (情境推荐) 当用户 u 在时间 t' 出现在位置 l' 时, 根据周边环境的 POI(point of interest) 信息和用户的历史行为兴趣点, 对用户进行推荐, 使其在相类似的情境 S 之下, 能够感受到与情境 S 类似的服务. 这就是情境由 S 迁移到 S' 时, 虽然时间和位置发生了变化, 但是服务却尽可能符合用户兴趣的情境推荐.

定义 6 (反馈与知识更新) 根据用户对情境推荐的评价结果对 PMR 架构的感知和存储进行反馈和知识更新, 如果用户满意推荐结果则感知和存储对情境的认知和记忆加强, 反之则减弱.

3.2 核心理想

PMR 模型架构的核心思想来源于对人脑认知世界行为的模拟, 可概括如下:

- (1) 自认知现实情境到给出推荐结果的 PMR 过程主要由感知、记忆和推荐 3 个核心模块组成;
- (2) 该过程对信息的处理大致分为快速处理 (感知)、精细处理 (存储) 和反馈更新 (推荐与结果比较) 3 个处理通路;
- (3) 该架构的 3 个流程分别对应于人脑对现实世界的分析、记忆和决策 3 个阶段.

其中, 感知、存储和推荐 3 个模块的实现思路可概括为:

图 2 基于 PMR 架构的兴趣点推荐流程图

Figure 2 POIs recommendation flow chart based on PMR framework

● 感知实现. 为了达到推荐兴趣点给用户的目的, 文中需要对时间信息、空间信息和文本信息进行感知. 通过分析这 3 类信息, 可以挖掘出用户在某时间出现在某地的历史行为信息. 获取这些信息后, 就可提取出用户出入这些位置的特征. 例如, 用户是白天去还是晚上去, 是工作日去还是周末去等. 根据不同用户的不同行为特征, 就可以为各种用户定制出满足其个性化要求的推荐服务.

● 存储实现. 通过感知阶段的分析后, 把 3 类信息通过筛选后进行规则编码, 存储在 PMR 架构的存储单元, 以备推荐阶段利用和反馈更新.

● 推荐实现. 推荐阶段可以通过感知阶段进行快速推荐, 也可通过存储阶段进行历史行为的综合分析推荐. 所谓快速推荐, 即通过用户最近若干次历史行为进行推荐. 这种推荐方式虽然执行效率高, 但由于参照依据有限, 因此其准确率很难保证. 综合分析推荐虽然执行效率低, 但其结合了用户所有的历史行为特征, 进行了 TOP 排序, 因此其准确率相对较高. 实际运用中, 可以把二者的优势相结合, 即设置推荐缓存单元, 把经过存储阶段 TOP 排序后的结果与用户最近几次的行为相结合存入缓存单元进行综合评估后给出推荐结果.

正如上文所述, 情境除了能够被感知、存储和推荐之外, 如果缺少反馈更新是不够的. 因此为了考虑该系统的主动学习性, 引入了反馈更新机制. 接下来, 将详细介绍情境感知、存储、推荐和反馈 (PMRF) 的工作流程和计算方法.

4 PMRF 计算

所谓 PMRF 计算, 即在 PMR 系统架构的基础上加入反馈 (feedback) 计算. 从首次录入用户行为信息到兴趣点推荐的迭代过程可用图 2 的流程图进行描述.

从图中可以看出, 基于 PMRF 系统推荐架构的流程分为感知、存储、推荐和反馈 4 个阶段. 其中, S_i 和 S_j 表示多种不同类型的外界刺激, P_i 和 P_j 表示在不同外界刺激下的多条不同的感知通路, M_i 和 M_j 表示对多种不同感知的存储, R_i 和 R_j 表示架构的多种推荐方案, 而 $0 \leq i \leq j \leq n$, 取值表示 i 和 j 分别是 n 种不同的处理流程. 因此, 架构的工作流程叙述如下:

- (1) 当系统感知到外界刺激 S_i 时, 把这种刺激传递给感知单元 P_i 进行处理, 分析这种刺激是已存在的刺激还是新刺激, 如果是已存在刺激则过滤, 否则就提取新刺激的特征进行编码;
- (2) 把新刺激传递给存储单元 M_i 进行存储, 以待该种刺激再次出现时进行情境的兴趣点推荐;
- (3) 当同种外界刺激再次出现时, 进行基于 PMR 架构的兴趣点推荐 R_i , 并做出行为 A_i 以待用户

Checkintime	Latitude	Longitude	Tweets
2010-03-03T05:56:13	-122.198	47.528139	A sprite can disappear in her mouth
2010-03-03T16:52:44	-122.198	47.528139	Lamo! I still get txt when AJ tweet
2010-03-03T16:57:24	-122.198	47.528139	Alright twitters tryna take me over
2010-03-03T20:20:40	-122.198	47.528139	Just got to work. Got my pizza bage
2010-03-03T23:23:33	-122.198	47.528139	Just got a txt from my cousin! Yes!
2010-03-03T23:37:36	-122.198	47.528139	Why is this woman in the bathroom?

图 3 实验数据样例

Figure 3 Experimental data sample

反馈. 如果用户反馈与行为相符, 即 $R_i = A_i$, 则该流程是符合用户兴趣点推荐的流程, 是一种基于特定用户兴趣的个性化推荐流程, 因此该流程权重被加强. 如果用户反馈与行为不相符, 即 $R_i \neq A_i$, 则该流程不符合用户的兴趣要求. 因此, 在系统中将把这种用户反馈作为一种新的外界刺激 S_j 被系统感知, 然后把它传递给感知单元作为一种新的感知 P_j 被分析和编码, 之后在存储单元被存储为 M_j , 作为一种新的推荐方案 R_j 以待推荐;

(4) 重复步骤 (1)~(3), 利用机器学习的思想不断更新和完善该系统, 以使其更加智能化.

4.1 感知计算

首先对感知信息进行分类. 本文要感知的数据信息主要为 3 类信息: 时间 (t)、位置 (l) 和文本 (r). 如图 3, 为文中需要感知数据信息的样例. 其中, 时间信息可按时间规则进一步划分为若干小类, 如年月日、时分秒、星期等, 兴趣点表现为 GPS 坐标 (对应相应的 POI 信息), 文本信息为用户评论.

从图 3 可以看出, 由于 PMR 架构的感知阶段需要感知的内容包括时间 t 、空间位置 l (GPS 坐标和 POI 信息) 和文本 r (微博内容), 因此以此 3 类信息为输入, 以感知情境 S 为输出, 可以将这 3 类信息表示为情境 S 的 3 阶张量形式, 用于描述感知映射过程,

$$S_{tlr} = S_t \times S_l \times S_r. \quad (1)$$

式 (1) 表示 3 个维度上以这 3 类信息分别为特征向量的情境信息. 由此, 可分别计算出某个维度上以某一类信息为特征向量的情境信息. 例如, 以时间信息为特征向量的情境信息可感知为

$$S_{lr}(t) = \int_T t S_{lr} dt, \quad (2)$$

其含义表征用户在时间段 T 内的位置信息和文本信息, 即活动轨迹和评论内容. 同理, 可表示出以位置信息和文本信息为特征向量的情境信息为

$$S_{tr}(l) = \int_L l S_{tr} dl, \quad (3)$$

$$S_{tl}(r) = \sum_R r S_{tl}, \quad (4)$$

式 (3) 表征用户在活动轨迹 L 范围内经历的时间信息和文本信息, 式 (4) 则从文本信息的角度进行了类似的表征. 其中式 (2) 和 (3) 中的 T, L 分别表示时间、位置可积范围, 由于式 (4) 中文本为离散量, 因此采用在范围 R 上的求和运算表示. 其求和的方法是, 以某用户对某条推荐的所有评论为求和区域 R , 在 R 内对评论的意图显著性首先采用文本相似度聚类, 然后对聚类进行评分, 评分方法与后文式 (19) 类似. 以单个词评分和词频为基础, 累积求和得到每条评论的得分, 进而利用累加评分的结果实

现对文本信息的张量求和. 文中假设时间和位置为矢量信息, 文本为标量信息. 因此, 赋予时间信息以大小和方向进行时间分片, 即以当前时间为参照基准 0, 过去和将来时间的边界可标记为 -1 和 1 , 那么介于边界 $(-1, 0)$ 之间为过去时间段, 介于 $(0, 1)$ 之间的为将来时间段, 因此, $T \subseteq (-1, 1)$. 同理, 以 GPS 位置为参照赋予位置信息以大小和方向, 以编码规则标记文本的大小. 由此, 可用张量来计算任意两两时间 (t, t') 、位置 (l, l') 和文本 (r, r') 之间的距离:

$$\begin{aligned} S_{\Delta tlr} &= S_t \times S_l \times S_r - S_{t'} \times S_l \times S_r, \\ S_{t\Delta lr} &= S_t \times S_l \times S_r - S_t \times S_{l'} \times S_r, \\ S_{tl\Delta r} &= S_t \times S_l \times S_r - S_t \times S_l \times S_{r'}. \end{aligned} \quad (5)$$

根据式 (5) 可感知情境信息的差异, 以识别、分析和存储不同的情境信息. 很显然, 若为同一情境信息则式 (5) 中的式子均取值为零. 为了把 GPS 坐标对与 POI 点统一起来, 可采用位置区域划分的思路^[22], 采用式 (6) 来解决问题.

$$\eta = \sqrt{\frac{\exp(\frac{-1}{2}(l_{i,0}^2 + l_{i,1}^2)) - \exp(\frac{-1}{2}(l_{i',0}^2 + l_{i',1}^2))}{\lambda(\sqrt{(l_{i,0} - l_{i,1})^2 + (l_{i',0} - l_{i',1})^2})}}, \quad (6)$$

其中, λ 为集合, 分母为 POI 点的分布区域, $l_{i,0}, l_{i,1}, l_{i',0}, l_{i',1}$ 为位置 l_i 和 $l_{i'}$ 的 GPS 坐标对, 计算得出 η 值与 POI 的信息点数进行比较, 如果 η 大于该区域内的 POI 信息点数, 则把该区域内的 GPS 坐标看作一点后与距离最近的 POI 点进行匹配; 如果 η 小于该区域内的 POI 信息点数, 则把离区域中心点最近的 GPS 坐标与离该坐标最近的 POI 信息点进行匹配.

4.2 存储计算

各类信息与编码存储之间的映射关系如下:

- 时间信息. 其编码规则为, 年 — 月 — 日 — 时 — 分 — 秒 — 星期. 以 2014 年 3 月 11 日星期二, 16:15 分为例, 按照编码规则就可被编码为 2014-03-11-16-15-02, 之后将该信息存储为二进制编码 11111011110-11-1011-10000-1111-10-XX, 其中 XX 为校验位, 以保证编码的正确性.

- 位置信息. 其在文中的数据集中分为两类, GPS 坐标信息和 POI 信息. 其中 GPS 坐标按照经度在前、纬度在后进行编码, 例如, GPS 坐标对为 (10.123, 158.021), 那么可编码为 10-123-158-021. 而 POI 信息按照地图数据分类编码表进行分类编码. 两类信息均转化为二进制编码进行存储.

- 文本信息. 其编码规则参照 GB2312 信息交换用汉字编码字符集进行二进制编码存储.

4.3 推荐计算

快速推荐. 以用户最近的若干次行为为参照进行粗略估计. 文中采用 N 阶 Markov Chain 模型进行预测推荐. 详细的计算方法可概括如下: 建立推荐模型的映射关系, 即 N 阶 Markov 状态空间, $\{X_1, X_2, \dots, X_n, X_{n+1}\}$, 其中假设 X_n 为当前状态, 它与感知信息的映射关系为

$$X_n = X(n) = S_{tlr} = S_t \times S_l \times S_r = \int_{t \in T} t S_{lr} dt \times \int_{l \in L} l S_{tr} dl \times \sum_{r \in R} r S_{tl}, \quad (7)$$

其状态空间概率分布的条件概率为

$$P(X_{n+1}|X_n) = \int P(X_{n+1}, X_n|X_{n-1}) dX_n = \int P(X_{n+1}|X_n) P(X_n|X_{n-1}) dX_n, n = 1, 2, 3, \dots, n. \quad (8)$$

因此, 利用式 (7) 和 (8) 可计算出第 $n+1$ 阶的状态空间的条件概率和感知信息. 从而给出快速推荐的方案. 通过分析上述公式可以发现, 快速推荐的效率和准确率与 n 值有关, n 值越大则执行效率越低, 准确率越高. 值得说明的是, 快速推荐的结果是针对单个用户时间序列上的个性化情境推荐, 因此该方法有其局限性.

综合推荐. 它是以用户所有的历史行为为参照进行综合评估, 然后建立 TOP 排序规则, 提取 TOP- n 进行推荐. 其实质是对用户的历史行为进行遍历计算的结果. 为了优化综合推荐的执行效率, 文中针对用户历史行为信息采用 Logistic 回归和期望最大化 (EM) 算法进行合并和检索 [23], 然后基于权重 ω 的 TOP- n 排序算法进行推荐. 接下来, 将介绍综合推荐的详细计算过程. 由于 Logistic 回归的方法能够得到较好的查询结果, 因此首先定义检索函数如下:

$$P(y_{tlr}^q = 1 | \omega, x_{tlr}^q) = \frac{1}{1 + \exp(-\omega \cdot x_{tlr}^q)} = \sigma(\omega \cdot x_{tlr}^q), \quad (9)$$

$$P(y_{tlr}^q = -1 | \omega, x_{tlr}^q) = 1 - P(y_{tlr}^q = 1) = \sigma(-\omega \cdot x_{tlr}^q), \quad (10)$$

其中, 式 (9) 和 (10) 中的上标 q 表示对 3 类信息的查询, 下标 tlr 表示一次查询返回的时间、位置和文本 3 类信息, x_{tlr}^q 表示 3 类信息的特征向量, ω 表示与 x_{tlr}^q 相关的权重集合. y_{tlr}^q 表示通过查询 q 可返回与查询相关的结果. y_{tlr}^q 的可能取值为

$$y_{tlr}^q = \begin{cases} 1, & \text{与查询 } q \text{ 相关,} \\ -1, & \text{与查询 } q \text{ 不相关.} \end{cases}$$

当给定与训练数据相关的情境 (即上文 4.1 小节中的情境 S) 个数 TLR 和查询次数 Q , 把式 (9) 和 (10) 结合起来可得到检索模型的似然函数如下:

$$L(w) = \prod_{q=1}^{|Q|} \prod_{tlr=1}^{|TLR|} P(y_{tlr}^q | \omega, x_{tlr}^q) = \prod_{q=1}^{|Q|} \prod_{tlr=1}^{|TLR|} \sigma(y_{tlr}^q \cdot \omega \cdot x_{tlr}^q), \quad (11)$$

式 (11) 通过最大 Logistic 似然函数的若干次迭代之后就可计算出权重 ω . 在查询过程中, 为了优化查询效率, 需要把查询的中间结果进行合并操作. 文中把 TLR 个情境分为 K 组, 并赋予其限制条件 $\sum_{k=1}^K \lambda_{tldr,k} = 1$, 其中 $tlr = 1, 2, \dots, |TLR|$. 为了使式子更简化, 文中仅通过一次查询后得到式 (9)~(11) 然后去掉式子中 y_{tldr}^q 和 x_{tldr}^q 的上标 q . 进一步用 $\sigma_{tldr,k}$ 代替 $P(y_{tldr}^q | \omega_k, x_{tldr}^q)$. 因此, 可简写为 $\sigma_{tldr,k} = \sigma(y_{tldr}^q \cdot \omega_k \cdot x_{tldr}^q)$. 其中, 权重集合 $\omega = \{\omega_k | k = 1, \dots, K\}$, 限制条件集合 $\lambda = \{\lambda_{tldr,k} | tldr = 1, \dots, |TLR|; k = 1, \dots, K\}$. 给定查询 q , 使得 $\theta = \{\omega, \lambda\}$, 其中 θ 是权重集合 ω 与其 k 阶集群相关的参数集合. 对于给定模型, 与查询相关的相似度函数可将式 (11) 简化为 $L(\theta) = \prod_{tldr=1}^{|TLR|} \prod_{k=1}^K \lambda_k \theta_{tldr,k}$. 但基于上述式子很难对其直接进行优化, 因此考虑引入期望最大化算法 (EM). 不妨设 z_{tldr} 为与情境 S 相关的 K 维潜在变量, 当且仅当 z_{tldr} 中仅有一个元素为 1 时, 其他 $K-1$ 个元素均为 0. 因此其限制条件满足 $\sum_{k=1}^K z_{tldr,k} = 1$, 其中, $tlr = 1, 2, \dots, |TLR|$, $z_{tldr,k}$ 表示 z_{tldr} 中的第 k 个元素. 然后可写出 z_{tldr} 先验分布为

$$P(z_{tldr}) = \prod_{k=1}^K \lambda_k z_{tldr,k}, \quad (12)$$

紧接着定义随机变量 $Z = \{z_{tldr} | tldr = 1, \dots, |TLR|\}$, 由于每个情境之间的相互独立性, 因此 Z 的先验是一个情境全集的连乘

$$P(Z) = \prod_{tldr=1}^{|TLR|} \prod_{k=1}^K \lambda_k^{z_{tldr,k}}. \quad (13)$$

由此可通过连乘写出基于情境全集模型的相似度函数为

$$P(\mathbf{X}, \mathbf{Y} | Z, \Theta) = \prod_{tlr=1}^{|TLR|} \left(\prod_{k=1}^K \sigma_{tlr,k}^{z_{tlr,k}} \right). \quad (14)$$

式 (13) 中 $\mathbf{X}$ 表示情境全集的特征向量, $\mathbf{Y}$ 表示情境全集的相关向量, 把式 (13) 和 (14) 结合起来, 可得出全条件概率相似度函数为

$$P(\mathbf{X}, \mathbf{Y}, Z | \Theta) = \prod_{tlr=1}^{|TLR|} \prod_{k=1}^K (\lambda_k^{z_{tlr,k}} \sigma_{tlr,k}^{z_{tlr,k}}). \quad (15)$$

对式 (15) 取对数, 可得到其对数形式为

$$\log P(\mathbf{X}, \mathbf{Y}, Z | \Theta) = \prod_{tlr=1}^{|TLR|} \prod_{k=1}^K z_{tlr,k} (\log \lambda_k + \log \sigma_{tlr,k}). \quad (16)$$

EM 算法分为 E-Step 和 M-Step 两个步骤, 对于 E-Step 需要计算 $P(\mathbf{X}, \mathbf{Y}, Z | \Theta)$ 的先验概率. 因此, 根据式 (15) 可得到下述关系

$$P(Z | \mathbf{X}, \mathbf{Y}, \Theta) = \frac{P(\mathbf{X}, \mathbf{Y}, Z | \Theta)}{P(\mathbf{X}, \mathbf{Y} | \Theta)} \propto \prod_{tlr=1}^{|TLR|} \prod_{k=1}^K (\lambda_k \times \sigma_{tlr,k})^{z_{tlr,k}}. \quad (17)$$

假设全体 Z_{tlr} 均独立, 基于情境 S 将式 (17) 的右边因式分解得到每个变量 $Z_{tlr,k}$ 的期望为

$$E[Z_{tlr,k}] = \frac{\sum_{z_{tlr,k} \in (0,1)} z_{tlr,k} (\lambda_k \prod_{tlr=1}^{|TLR|} \sigma_{tlr,k})^{z_{tlr,k}}}{\sum_{z_{tlr,j}, 0 \leq j \leq 1} z_{tlr,j} (\lambda_j \prod_{tlr=1}^{|TLR|} \sigma_{tlr,j})^{z_{tlr,j}}} = \frac{\lambda_k \prod_{tlr=1}^{|TLR|} \sigma_{tlr,k}}{\sum_{j=1}^K \lambda_j \prod_{tlr=1}^{|TLR|} \sigma_{tlr,j}} = \gamma(z_{tlr,k}). \quad (18)$$

在 M-STEP 中, 根据式 (18) 可计算参数 $\Theta^n = \arg \max \psi(\Theta, \Theta^0)$, 其中 $\psi(\Theta, \Theta^0) = E[\log P(\mathbf{X}, \mathbf{Y}, Z | \Theta)]$, 结合式 (16) 可得到 M-STEP 的目标函数如下:

$$\{\lambda^n, \omega^n\} = \arg \max_{\lambda, \omega} \sum_{tlr=1}^{|TLR|} \sum_{k=1}^K \gamma(z_{tlr,k}) (\log \lambda_k + \log \sigma_{tlr,k}). \quad (19)$$

由式 (19) 可知, 只需使公式左边的前半部分最大化, 即 $\lambda^n = \arg \max_{\pi} \sum_{tlr=1}^{|TLR|} \sum_{k=1}^K \gamma(z_{tlr,k}) \log \lambda_k$ 使其满足限制条件 $\sum_{k=1}^K \lambda_k = 1$ 即可. 利用拉格朗日乘子并设置其梯度为 0, 能得到 λ_k 的优化值为 $\lambda_k^n = \frac{\sum_{tlr=1}^{|TLR|} \gamma(z_{tlr,k})}{\sum_{k=1}^K \sum_{tlr=1}^{|TLR|} \gamma(z_{tlr,k})}$, 至此可得出下述公式

$$\omega^n = \arg \max \sum_{tlr=1}^{|TLR|} \sum_{k=1}^K \gamma(z_{tlr,k}) \log \sigma_{tlr,k} \quad (20)$$

来解决推荐方案的权重得分优化问题. 至此, 通过上述一系列计算过程就可以得到权重集合 Θ , 它就是每种推荐方案的初始得分.

快速综合推荐. 通过建立缓存区域, 结合考虑上述两种推荐方法. 缓存区内首先利用快速排序建立推荐的初始 TOP- n 排序队列, 然后针对队列中的推荐方案利用综合推荐计算推荐方案得分, 紧接着根据得分从高到低的顺序调整 TOP- n 队列的排序, 由此得到初始排序. 其优势在于既提高了执行效率, 又提高了推荐的准确率. 由于计算方法类似, 此处不再赘述.

4.4 反馈计算

反馈计算主要是基于用户评论反馈的计算, 针对用户发表评论进行处理后提取结果对用户推荐方案进行更新. 因此, 利用线性回归的思想建立线性回归模型用于反馈计算^[24,25]. 在此之前, 首先需要对文本信息进行预处理, 现介绍针对用户评论文本信息的处理规则如下:

- 去除所有 HTML 标记;
- 过滤掉包含 5 个以上的非 ASCII 码字符的评论;
- 用分隔符把评论分开, 分隔符包括空格、逗号、冒号等;
- 把所有的英文字母转为小写;
- 去除不常见的词汇, 例如在样本总体中词频小于 5 次的词汇;
- 提取表明立场的词汇建立词汇库. 满意推荐的词汇标注为正, 不满意推荐标注为负, 中立态度标注为 0, 正负的标注程度经过用户打分获取. 为了实验的方便, 文中不妨用标注分值取值范围度量用户对推荐的满意程度. 经过以上规则的文本预处理过程之后, 就得到了用户评论的样本总体和立场词汇库. 接下来, 建立反馈递归模型进行反馈计算. 文中采用线性递归模型建立评论内容与推荐方案之间的计分规则关系.

$$\text{Level}(W) = \sum_{i=1}^n (Y_i + (\omega_0 + \sum_{j=1}^m X_{ij}\omega_{ij}))^2, \quad (21)$$

其中, $\omega = \{\omega_0, \omega_1, \dots, \omega_n\}$ 表示词汇的权重集合, Y_i 用于表示评论中针对第 i 个子句的评分, ω_0 表示评论中第一个词汇的权重, 此处用于尽量避免评分等级为零的情况. X_{ij} 用于统计评论中第 i 个子句中第 j 个词汇的词频, ω_{ij} 表示评论中第 i 个子句中第 j 个词汇的得分权重. 通过式 (21) 的计算得到 $\text{Level}(W)$, 它就是反馈计算得到的反馈得分. 然后将其反馈给感知单元和存储单元, 把反馈得分与推荐方案的初始 TOP- n 得分进行累加求和计算总得分, 以此达到对推荐方案排序进行更新的目的. 值得说明的是, 由于用户反馈有正负面之分, 因而计算的反馈得分也可能取负值, 进而可能使推荐方案的 TOP 排名下降, 这样可以使 PMR 架构更符合用户的个性化特征.

5 实验

本章将首先在 5.1 小节中给出实验所用的公用数据集, 在 5.2 小节中将给出实验的评估标准, 以验证算法的实际性能. 然后, 在 5.3 小节中将给出兴趣点推荐的主要实验结果, 并利用 5.2 小节中提出的标准予以评估, 并进行实验结果分析.

5.1 数据集

文中的实验采用两个数据集, 其中 Geo-Text 为公用数据集, Sina-Tweets 是利用 Sina API 从 Sina Micro-blog 上爬取的 2012 年 12 月 1 日到 2013 年 7 月 31 日的 20 多万条数据, 各数据集所包含的详细内容如表 1 所示. 表 1 中的数据集都经过了预处理, 剩下的都是每天至少有 5 个访问位置的活跃用户. 其中, 用户微博评论内容按照 4.4 小节中论述的数据处理规则进行处理.

5.2 评估标准

实验将从兴趣点推荐的算法执行效率和实际效果两方面进行评估.

表 1 实验数据集

Table 1 Experimental data-sets

Data sets	Data items	Number	Description
Geo-Text	User ID	9481	Number of users
	Check-in time	377616	Time of check-ins
	GPS coordinates	54325	Position Info
	Tweets	377616	Number of tweets
Sina-Tweets	User ID	20106	Number of users
	Check-in time	197627	Time of check-ins
	GPS coordinates	61563	Position Info
	Tweets	199508	Number of tweets

表 2 满意等级与评分的关系

Table 2 Relationship of RSI and ratings

Ratings	VS	WS	QS	QDS	WDS	VDS
Score	6	5	4	3	2	1

(1) 效率评估. 召回率 $\text{Recall}@n$: 评估 4.3 小节推荐计算中综合推荐搜索算法的查询效率 [26~29]. 计算过程如下, 由于综合推荐的最终结果需要提取 TOP- n 个推荐方案给用户选择, 因此首先在测试集中定义 $\text{hit}@n$, 作为命中标识. 如果测试集中被标记为推荐的样本出现在推荐集合 TOP- n 中, 那么 $\text{hit}@n$ 取值为 1, 否则取值为 0. 计算规则如式 (22),

$$\text{Recall}@n = \frac{\text{hit}@n}{|S_{\text{test}}|}, \quad (22)$$

其中, $\text{hit}@n$ 表示测试集中被命中的样本数, $|S_{\text{test}}|$ 表示所有测试集中的样本总数.

(2) 效果评估. 推荐满意度 (recommendation satisfaction index, RSI). 度量用户对某推荐的满意程度. 描述为在对某推荐的满意度测评中, 对该推荐表示满意的样本数 (SP) 与测评总人数 (ST) 的比值, 如式 (23).

$$\text{RSI}_j(i) = \frac{\text{SP}_{ij}}{|\text{ST}_j|}, \text{RSI} = \frac{\sum_{j=1}^m \sum_{i=1}^n \text{RSI}_j(i)}{\sum_{j=1}^m \text{ST}_j}, \quad (23)$$

其中, SP_{ij} 表示第 j 个测试集中用户对第 i 条推荐表示满意的样本数, $|\text{ST}_j|$ 表示第 j 个测试集的测评总人数, m 为测试集总数, i 为推荐数总数. 得出中间结果 $\text{RSI}_j(i)$ 就是第 j 个测试集中用户对第 i 条推荐的满意度. 再求该结果在所有测试集中的均值计算 RSI, 最后把 RSI 作为反馈传递给系统进行推荐评分, 以作为情境感知与存储更新的重要依据. 因此, 文中不妨设用户对推荐结果的满意程度分为 6 个等级. 非常满意 (VS)、很满意 (WS)、比较满意 (QS)、比较不满意 (QDS)、很不满意 (WDS) 和非常不满意 (VDS), 且设它与评分的关系为式 (24), 如表 2, 不低于 3 分统计为用户满意, 不高于 3 分统计为用户不满意.

推荐准确率 (recommendation accuracy rate, RAR). 表征推荐系统的推荐与用户反馈的契合程度. 它与 RSI 的区别在于, RSI 是仅针对单个推荐做出的评估, 而 RAR 是针对推荐样本的评估. 文中利用

表 3 两两用户关系与 k 值计算结果Table 3 Relationship of users and k values

User relationship	1,2	1,3	1,4	2,3	2,4	3,4
k	0.701	0.782	0.736	0.821	0.681	0.768

表 4 用户组与标注样本数的关系

Table 4 Relationship of groups and samples

User group ID	1	2	3	4
Labeled tweets	20000	20000	20000	20000

正则化折扣累积收益 ($nDCG$) 来度量推荐精度并给出计算公式,

$$DCG = rel_1 + \sum_{i=2}^n \frac{rel_i}{\log_2 i}, nDCG = \frac{DCG}{IDCG}, \quad (24)$$

其中, rel_i 是推荐系统对第 i 个推荐的评分, 因此用 DCG 计算出的是 PMR 计算得出的折扣累积收益, 而 IDCG 为用户评分给出的折扣累积收益, 因此二者的比值 $nDCG$ 表达的含义正好就是推荐系统与用户意愿的契合程度, 即推荐准确率.

(3) 用户标注一致性. 由于用户对推荐结果评分必须通过用户完成, 因此势必造成结果的主观性. 为了说明其合理性, 文中对评分结果采用科恩卡帕系数 (Cohen's Kappa)^[30,31] 进行一致性评测. 计算公式为 $k = \frac{Pr(a)-Pr(e)}{1-Pr(e)}$. 计算结果如表 3, 可以看出绝大部分 k 值均介于 0.6~0.8 之间, 且 $k(2,3)=0.821$ 介于 0.8~1 之间. 由于, $k \in (0.6, 0.8]$ 达到适中一致性, $(0.8, 1]$ 达到完美一致性, 因此可以认为 4 个用户的评价具有专业性和准确性. 值得说明的是, 此处的样本标注仅为 4 组用户标注的测试集样例, 实际实验中会按照一定比例将训练集和测试集进一步拆分为小样本集合 (如 8:2 计 10 份或 16:4 计 20 份的样本比) 进行实验.

5.3 实验结果与分析

文中实验以最新的协同过滤位置推荐模型 CLR 架构为标准形成对比实验. 由于该架构原型实验评估标准中仅推荐准确率一项与文中评估标准一致, 不曾涉及召回率与满意度的评估, 因此只能在推荐准确率上与该模型形成对比.

在给出实验结果前, 首先要对推荐系统的 TOP- n 进行设置说明, 即随着 n 值的变化, 上文 5.2 小节中评估标准公式的计算结果也会随之变化. 为了简化同性质实验的工作量, 仅提取表 4 中的一个用户组作为训练集合, 余下的数据用作测试集合以验证系统性能. 值得注意的是, 测试集是对训练集的扩充, 其实验条件和配置与训练集完全相同. 因此, 通过训练集训练系统模型, 测试集验证了系统性能后, 便得出了实验的最终结果. 值得说明的是, 推荐准确率的取值是取快速推荐、综合推荐和快速综合推荐 3 者中 TOP- n 排序的比较最终结果作为评估标准衡量的最终展示结果.

(1) 召回率. 召回率描述的是被标记样本出现在推荐集合 TOP- n 中的数目占测试集中被标记为推荐样本数目的比例. 实验结果如图 4.

图 4 中描述了快速综合推荐、综合推荐和快速推荐这 3 种不同推荐方法分别在两个不同的数据集 Sina-Tweets 和 Geo-Text 上的召回率对比情况. 从图中可以看出, 随着命中数的增大, 召回率逐渐增大, 即随着测试集的增大, 被标记为推荐的样本数随之增加, 当 TOP- n 集合 (即推荐需求增大时),

图 4 (网络版彩图) 推荐查询召回率

Figure 4 (Color online) Recall@n

图 5 (网络版彩图) 测评用户数与 RSI

Figure 5 (Color online) Test users number and RSI

图 6 (网络版彩图) 推荐数与 RAR

Figure 6 (Color online) Recommended number and RAR

只要被标注为推荐的样本就基本都能被加入 TOP- n 集合, 因而导致了召回率的逐步增大, 但由于系统存在反馈功能, 因而导致了实验结果与理论分析存在不同, 因此召回率并不能达到 100%, 并且有些样本数取值处出现了召回率高于或低于理论值 (如图 4 中上下波动的图形) 的情形。

(2) 测评用户数与 RSI. RSI 描述的是对某推荐进行满意度测评中, 满意人数占测评用户数的比例。其实验结果如图 5 所示。图 5 中描述了快速综合推荐、综合推荐和快速推荐这 3 种不同推荐方法分别在两个不同的数据集 Sina-Tweets 和 Geo-Text 上的 RSI 对比情况。从图中可以看出, 大部分测试用户对推荐结果都比较满意, 甚至有部分用户对结果很满意。随着测试用户数的增加, 用户满意度在样本人数低于 450 时, 出现急剧下降和上升的情形, 当样本人数高于 450 时, 其用户满意度趋于平稳 (在 60% 左右波动)。其原因在于, 当测试人数较少时, 由于人的主观因素在测评小样本中无法避免, 因而会出现满意度评估客观性较差的情况, 但是随着样本数逐步增加, 部分样本的主观倾向被大样本采样摒除, 因此用户满意度的测评可信度在大样本中趋于客观。

(3) 推荐数与 RAR. RAR 描述的是用户表示满意的推荐数目占系统推荐数目的比例, 以此证明系

图 7 (网络版彩图) 样本数与 TOP- n 得分Figure 7 (Color online) Samples and TOP- n ratings

统与用户契合程度. 其实验结果如图 6 所示.

从图 6 中可以看出, PMR 架构在 Sina-Tweets 和 Geo-Text 两个数据集上的推荐准确率均略高于 CLR 架构 (以最高推荐准确率进行比较, Sina-Tweets 上 PMR 架构为 87.27%, CLR 架构为 81.82%; Geo-Text 上 PMR 架构为 81.79%, CLR 架构为 80.21%). 随着推荐数的增大, 推荐准确率最初出现急剧上升和下降的情况, 当推荐数增大到 500 左右后, 准确率趋于平衡 (在 70% 左右波动). 其原因与 RSI 的测评类似, 同样都是因为人的主观标注在小样本采样中存在主观倾向的现象, 而这一点也正好证明了本文实验采样的合理性和样本标注的真实性.

(4) 样本数与 TOP- n 得分. 如图 7 所示, 为实验推荐方案样本数与该方案的 TOP- n 总得分的结果. 其中, 横坐标表示样本随机编号, 纵坐标表示样本的 TOP- n 得分值. 12 个子图中分别展示了样本数从 10 增加到 1000 时, 推荐方案得分的取值情况. 通过结合图 6 与图 7 分析可知, 样本数较少时, 由于用户可评估样本少, 因而推荐准确率不高, 但随着样本数增加, 可供选择样本数逐渐增加导致用户可选的推荐方案增加, 因而推荐准确率呈现上升趋势, 但随着样本数进一步增加时, 由于受到限制条件集合 λ , TOP- n 排名、快速综合推荐缓存区队列等多方面因素的限制, 推荐准确率通过迭代后逐渐趋于稳定.

(5) 算法复杂度和算法处理时间比较. 对比实验选取的 baseline 模型是 CLR 架构^[20], 它的推荐过程的模型描述如图 8 所示. 图中的 CADC(U) 表示被与用户 u_a 相似的用户访问过的位置集合, PTN(U), NRM(U) 和 TRL(U) 表示分别被与 u_a 相似的模式用户、正常用户和旅行用户访问过的位置集合, PTN(U, A) 和 PTN(U, L) 分别表示满足活动、模式用户相似的位置集合和满足位置相似、模式用户相似的位置集合, NRM(U, A) 和 NRM(U, L) 分别表示满足活动、正常用户相似的位置集合和满足位置相似、正常用户相似的位置集合, TRL(U, A) 和 TRL(U, L) 分别表示满足活动、旅行用户相似的位置集合和满足位置相似、旅行用户相似的位置集合, PTN(U, A, L), NRM(U, A, L) 和 TRL(U, A, L) 表示分别表示满足活动、位置、模式用户相似的位置集合, 满足活动、位置、正常用户相似的位置集合, 满足活

图 8 CLR 架构推荐模型

Figure 8 Recommendation model of CLR framework

图 9 (网络版彩图) 推荐方案数与处理时间

Figure 9 (Color online) Number of recommendations and processing time

动、位置、旅行用户相似的位置集合. 从图 8 中的模型描述可知, $PTN(U)$, $NRM(U)$ 和 $TRL(U)$ 分别要对用户进行遍历, 算法复杂度为 $O(n)$. $PTN(U,A)$ 和 $PTN(U,L)$, $NRM(U,A)$ 和 $NRM(U,L)$, $TRL(U,A)$ 和 $TRL(U,L)$ 分别要对用户和活动、用户和位置进行二维度遍历, 算法复杂度为 $O(n^2)$. $PTN(U,A,L)$, $NRM(U,A,L)$ 和 $TRL(U,A,L)$ 分别要对用户、活动和位置进行三维度遍历, 算法复杂度为 $O(n^3)$. 因此, 整个算法的复杂度为 $O(n^3)$. 如图 9 所示, PMR 架构与 CLR 架构在两个不同的数据集 Sina-Tweets 和 Geo-Text 上表现出差异较明显的算法处理时间. 随着推荐方案数目的增加, PMR 架构的推荐算法由于具有期望最大化 (EM) 算法且进行了函数平滑度处理, 所以它始终都能求得推荐方案数中的极优解且时间复杂度被取对数降低. 因此, PMR 架构的算法复杂度为 $O(\log n)$. 其算法处理时间曲线与 CLR 架构相比, 算法处理时间更加稳定. 而 CLR 架构给出的推荐算法中由于存在多个循环, 当推荐方案数目过大时, 其算法由于循环次数的持续增加就会导致算法处理时间的持续上升.

6 结论

本文通过对 PMJ 模型的思想借鉴建立一种全新的位置推荐系统原型 PMR 架构, 它能够对用户的历史行为进行挖掘, 并结合最新的情境感知、存储、推荐和反馈机制建立了一整套最新的情境计算的计算方法, 该架构解决了基于位置的移动社交网络研究中利用情境进行实时兴趣点推荐的问题. 其最大的优势和创新点在于, 它能对用户反馈信息进行实时的处理和对推荐方案的及时更新, 通过指定用户的反馈计算推荐方案针对用户的个性化得分, 使推荐方案能够针对用户的兴趣自适应修正, 以此更加满足用户需求. 实验通过对 PMR 的召回率、用户满意度和准确率的评估结果证明, 该系统的性能在解决兴趣点推荐问题时, 能够达到较高的召回率和准确率, 并且能够较好地契合用户的实际需求. 在将来的研究工作中, 将结合情境的其他重要因素, 如 5W(何人 Who+, 何时 When+, 何地 Where+, 何事 What+, 何原因 Why) 等, 以求更加深入地提升系统的性能.

参考文献

- 1 Noulas A, Scellato S, Lathia N, et al. A random walk around the city: new venue recommendation in location-based social networks. In: *Proceedings of Social-Com*, Amsterdam, 2012. 144–153
- 2 Cho E, Myers S A, Leskovec J. Friendship and mobility: user movement in location-based social networks. In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, 2011. 1082–1090
- 3 Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans Knowl Data Eng*, 2005, 17: 734–749
- 4 Linden G, Smith B, York J. Item-to-item collaborative filtering. *Internet Comput*, 2003, 7: 76–80
- 5 Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms. In: *Proceedings of the 10th International Conference on World Wide Web*, Hong Kong, 2010. 285–295
- 6 Su X, Khoshgoftaar T M. A survey of collaborative filtering techniques. *Adv Artif Intell*, 2009, 2009: 4
- 7 Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks. In: *Proceedings of the 4th ACM Conference on Recommender Systems*, Barcelona, 2010. 135–142
- 8 Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Comput*, 2009, 42: 42–49
- 9 Vozalis M G, Markos A, Margaritis K G. Collaborative filtering through SVD-based and hierarchical non-linear PCA. In: *Proceedings of ICANN*, Thessaloniki, 2010. 395–400
- 10 Ye M, Yin P, Lee W C. Location recommendation for location-based social networks. In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, San Jose, 2010. 458–461
- 11 Pham M C, Cao Y, Klamra R, et al. A clustering approach for collaborative filtering recommendation using social network analysis. *J Univ Comput Sci*, 2011, 17: 583–604
- 12 Yildirim H, Krishnamoorthy M S. A random walk method for alleviating the sparsity problem in collaborative filtering. In: *Proceedings of the 2008 ACM Conference on Recommender Systems*, Lausanne, 2008. 131–138
- 13 Konstas I, Stathopoulos V, Jose J M. On social networks and collaborative recommendation. In: *Proceedings of SIGIR*, Boston, 2009. 195–202
- 14 Scellato S, Noulas A, Mascolo C. Exploiting place features in link prediction on location-based social networks. In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Diego, 2011. 1046–1054
- 15 Bao J, Zheng Y, Mokbel M F. Location-based and preference-aware recommendation using sparse geo-social networking data. In: *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, Redondo Beach, 2012. 199–208
- 16 Levandoski J J, Sarwat M, Eldawy A, et al. Lars: a location-aware recommender system. In: *Proceedings of ICDE*, Washington, 2012. 450–461
- 17 Fu X L, Cai L H, Liu Y, et al. A computational cognition model of perception, memory, and judgment. *Sci China Inf Sci*, 2014, 57: 032114

- 18 Zhang Y, Guo Q, Zhang L, et al. Deep web and cognitive computing. *Commun CCF*, 2014, 10: 26–32 [章毅, 郭泉, 张蕾, 等. 深度网络和认知计算. *中国计算机学会通讯*, 2014, 10: 26–32]
- 19 Yin H Z, Sun Y Z, Cui B, et al. LCARS: a location-content-aware recommender system. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, 2013. 221–229
- 20 Leung K W T, Lee D L, Lee W C. CLR: a collaborative location recommendation framework based on co-clustering. In: *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Beijing, 2011. 305–314
- 21 Dempster A P, Laird N M, Rubin D B. Maximum likelihood from incomplete data via the em algorithm. *J Roy Stat Soc*, 1977, 39: 1–38
- 22 Ma H, Lyu M R, King I. Learning to recommend with trust and distrust relationships. In: *Proceedings of the 3rd ACM Conference on Recommender Systems*, New York, 2009. 189–196
- 23 Fu B, Lin J L, Li L. Why people hate your app—making sense of user feedback in a mobile app store. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, 2013. 1276–1284
- 24 Fornell C, Johnson M D, Anderson E W, et al. The american customer satisfaction index: nature, purpose, and findings. *J Market*, 1996, 60: 7–18
- 25 Cremonesi P, Koren Y, Turrin R. Performance of recommender algorithms on top-n recommendation tasks. In: *Proceedings of the 4th ACM Conference on Recommender Systems*, Barcelona, 2010. 39–46
- 26 Chen W Y, Chu J C, Luan J, et al. Collaborative filtering for orkut communities: discovery of user latent behavior. In: *Proceedings of the 18th International Conference on World Wide Web*, Madrid, 2009. 681–690
- 27 Koren Y. Factorization meets the neighborhood: a multi-faceted collaborative filtering model. In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Las Vegas, 2008. 426–434
- 28 Yin H, Cui B, Li J, et al. Challenging the long tail recommendation. In: *Proceedings of VLDB Endowment*, Istanbul, 2012. 896–907
- 29 Jacob C. A coefficient of agreement for nominal scales. *Educ Psychol Meas*, 1960, 20: 37–46
- 30 Cohen J. Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. *Psychol Bull*, 1968, 70: 213–220
- 31 Yuan Q, Cong G, Ma Z, et al. Who, where, when and what: discover spatio-temporal topics for twitter users. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, 2013. 605–613

Study of POI-s recommendation based on a PMR framework

WANG Feng, YU Wei* & LI ShiJun

Computer School, Wuhan University, Wuhan 430072, China

*E-mail: afengcom@163.com

Abstract Location-based mobile social networks, such as micro-blogs, web-chats, etc. have provided mass historical user behavior information such as user profiles, check-in times, position tracking, and tweets, for research on mobile social networks. The availability of these data received from users offers a good opportunity to study the users' real-time POI-s recommendation based on context. In this paper, a new framework for location recommendation called PMR is enhanced by drawing from the idea of a PMJ model. One of the greatest strengths of the prototype is its ability to process users' feedback information in real time and update recommendations by integrating the latest methods of context awareness and the strength of a PMJ framework into a PMR framework. Additionally, it can re-score the recommended scheme and update the rank of the scheme so that it fits users' demands. Based on this, a set of algorithms are proposed to solve the key problems in the research of context-aware computing: context awareness, context storage, context recommendation, and context feedback. Experimental results show that the model performs with a higher recall rate (96.93%) and precision rate (87.27%) and a better

customer satisfaction index (80.12%).

Keywords location-based social network, PMR framework, POI-s recommendation, context awareness, urban computing

WANG Feng was born in 1984. He is a Ph.D. candidate at the School of Computers, Wuhan University, and a student member of the CCF. His research interests include mobile social networks and social media.

YU Wei was born in 1987. He received his Ph.D. degree in computer applications from the School of Computers, Wuhan University, in 2011. He is a current member of the CCF. His research interests include web data mining, web community, virtual users, social networks, web data evaluation, and big data on the web.

LI ShiJun was born in 1964. He is Professor at the School of Computers, Wuhan University, and a senior member of the CCF. His research interests include web search and mining, management and optimization of mass data, data mining, database technology, and Android mobile development.