

使用轨迹指纹和地点相似性的地点推荐

印桂生 程伟杰 董宇欣 董红斌 张万松

(哈尔滨工程大学 计算机科学与技术学院 黑龙江 哈尔滨 150001)

摘要: 针对传统的时空轨迹相似性度量算法中存在的计算复杂度高且不适于增量计算的问题,提出了基于相似哈希计算用户时空轨迹相似度的方法,同时使用逆轨迹频率(ITF)度量位置流行度对轨迹相似性的影响,将用户的历史轨迹编码为二进制轨迹指纹,并根据海明距离判断轨迹指纹之间的相似性,使得相似性计算可以在线性时间内完成;此外,改进了地点相似性算法,并将轨迹相似度和地点相似度相结合提出了基于地点和轨迹相似性的地点推荐算法。实验结果表明,本文的推荐方法在准确率、召回率和覆盖率方面能够取得较好的推荐效果,验证了所提方法的有效性。

关键词: 时空轨迹; 轨迹相似性; 轨迹指纹; 地点流行度; 地点推荐

doi: 10.11990/jheu.201506035

网络出版地址: <http://www.cnki.net/kcms/detail/23.1390.u.20151218.1051.004.html>

中图分类号: TP391 **文献标志码:** A **文章编号:** 1006-7043(2016)03-414-06

Location recommendation using trajectory fingerprints and location similarity

YIN Guisheng, CHENG Weijie, DONG Yuxin, DONG Hongbin, ZHANG Wansong

(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)

Abstract: In order to solve the problem of the high computational complexity and inapplicability to incremental computing of traditional spatial-temporal trajectory similarity measurements, in this paper we propose a simHash-based method to measure the similarity between different users' spatial-temporal trajectories, which also consider the influence of location popularity on the trajectories' similarities by using the locations' inverse trajectory frequency (ITF). With this method, users' trajectories are initially transformed into binary trajectory fingerprints. We use the Hamming distance to determine the similarity of the users' trajectories, and the similarity calculation can be finished within linear time. In addition, we propose an improved location similarity algorithm and combine the location similarity with the trajectory similarity to generate interesting location recommendations. Compared with the existing method, the experimental results verify the effectiveness of the proposed method and demonstrate that it has better performance with respect to precision, recall, and coverage.

Keywords: spatial-temporal trajectory; trajectory similarity; trajectory fingerprint; location popularity; location recommendation

随着位置定位技术和无线通讯技术的发展,智能联网设备将位置定位技术与移动网络技术相结合,为用户提供了便利且无时无刻和无处不在的移动服务,这就使得收集反映用户行为规律和兴趣习惯的日常轨迹成为可能。通过移动网络技术收集的时空轨迹数据可以用于分析用户的移动特点和生活规律,从而深入理解用户、位置以及两者之间的相关

关联^[1]。从最早的地理信息系统到现在的移动网络服务,研究人员基于用户历史轨迹做了大量分析和地理的研究^[2]。在基于历史轨迹理解用户和地理空间的研究中,判断地理轨迹的相似性^[3]成为一个关键的问题。因为相似用户更容易具有相似的行为偏好,而基于相似用户向用户推荐服务也是协同过滤的主要思想。因此,通过分析大量用户的历史轨迹^[4],可以寻找与目标用户的历史轨迹相似的用户,从而可以为用户提供基于位置的推荐服务,例如推荐饭店、推荐旅行线路或者推荐朋友等。

传统的基于轨迹的相似性计算都采用最长公共子串^[5-6]计算用户历史轨迹的相似性。该方法是极其费时的,尤其是历史轨迹较长时问题会变得更加

收稿日期: 2015-06-11.

网络出版日期: 2015-12-18.

基金项目: 国家自然科学基金资助项目(61272186, 61472095); 黑龙江省自然科学基金资助项目(F201110); 中央高校基础研究基金资助项目(HEUCF100604); 黑龙江博士后基金资助项目(LBH-Z12068).

作者简介: 印桂生(1964-),男,教授,博士生导师.

通信作者: 印桂生, E-mail: yinguisheng@hrbeu.edu.cn.

严重,此外,轨迹的改变和增加都需要重新计算相似性,而在基于位置的社会网络中,个人的时空轨迹是快速增加并且不断演化的,这就使得计算成本急剧增加。在已有的计算轨迹相似性的方法中,为了节约时间和计算资源,不得不加入公共子串长度的最大阈值来停止最大公共子串的检测^[5]。因此,在分析基于位置的社会网络中的时空轨迹的时候迫切需要一种快速且适合海量数据和增量计算的相似度计算方法。

相似哈希^[7-8]是局部敏感哈希的一种,由 Moses Charikar^[7]在2002年提出,是一种用单个哈希函数得到文档最小哈希签名,根据签名计算文档的相似性,去除重复网页的方法。本文根据人们的地理轨迹的计算特点,考虑地理位置在用户的地理轨迹中的重要性,将比较历史轨迹的相似性转换为对历史轨迹的签名的比较,将公共子串计算转换成二进制的异或比较。

此外,在给用户提供地点的时候,还考虑了地点之间的相似度,通过综合考虑用户习惯的相似度和地点之间的相似度,提出了混合轨迹和地点相似性的推荐算法。

综上,本文的工作主要是从GPS日志中提取用户的有效停留点,然后通过聚类停留点获取用户的一致可比较的用户轨迹历史,进而使用相似哈希获取轨迹的二进制指纹,将二进制指纹之间的海明距离作为用户的历史轨迹之间的相似性。综合考虑轨迹相似性和地点相似性,为用户提供地点推荐服务。

1 轨迹建模

一般GPS设备接收信号的平均周期比较短,这就造成GPS日志中记录的GPS点非常密集,如果直接用于计算,将导致极大的计算量;此外,由于GPS日志中的GPS点的精度比较高,用户在同一地点会产生多个GPS点,如果直接使用GPS点计算用户的相似性,就认为2个用户没有访问共同的地点,事实上2个用户访问的是同一个地点。因此,有必要将GPS日志中的GPS点进行处理,提取能够表示用户生活习惯的有效停留点,并进一步使用基于密度的聚类算法,将所有用户的停留点聚类形成可比较的停留区域。停留区域不仅可以为用户提供可比较的轨迹数据,同时可以获取更丰富的语义信息。下面将简单介绍GPS日志的基本处理,包括停留点提取和停留区域聚类,具体的方法可以参考文献^[9]。

1.1 轨迹历史表示

GPS日志中的GPS坐标一般包含经纬度和采集时间等信息,首先需要从GPS日志中提取GPS坐标点,形成GPS轨迹历史。这里只关注GPS坐标点

的经纬度坐标和采集时间,将包含经纬度和采集时间的数据称为GPS坐标点。GPS坐标点可以表示成 $p = (a, o, t)$,其中 a 为纬度, o 为经度, t 为采集的时间。

如果将用户的GPS日志记录的GPS历史按照自然天划分,可以发现用户每天的轨迹几乎都以家为起点和终点,因此,可以将用户每天的日志历史记为用户的一条GPS轨迹,而用户的轨迹历史就是所有轨迹的集合。本文定义,用户的GPS轨迹是按照时间排列的GPS坐标点的序列,表示为 $s_u = \{p_1, \dots, p_i, \dots, p_m\}$,其中 p_i 是GPS坐标点, u 是用户, m 是指轨迹中的GPS坐标点的个数。用户的GPS轨迹历史是用户的所有GPS轨迹的集合,表示为 $h_u = \{s_u^i \mid u \in U\}$,其中 U 是所有用户的集合, $1 \leq i \leq n$, n 为GPS轨迹的数量,也就是用户记录的GPS轨迹的天数。

1.2 停留点提取

用户的轨迹历史中GPS坐标点比较密集,并且包含大量不能反映用户行为兴趣和特点的噪声数据,不仅增加了计算量,而且干扰了对用户真实行为的分析,因此需要从历史轨迹中提取能真实反映用户行为兴趣和特点的GPS坐标点,并使用这些点表示用户的历史轨迹。如果用户在某个区域内停留超过了一定时间,可以认为该用户在该区域内的GPS坐标点属于有效坐标点,可以使用该区域内所有GPS坐标点经纬度的平均值作为该区域的停留点坐标,进入该区域的时间戳作为该停留点的开始时间,离开该区域的时间戳作为该停留点的离开时间。

1.3 停留点聚类分析

在完成对每个用户的停留点提取工作后需要对停留点进行聚类分析,因为停留点是虚拟的坐标点,与用户的实际行为相差很大,如果直接对停留点分析,则分析的准确性大大降低。因此需要将所有用户的停留点进行聚类,形成一致的可比较的停留区域。在停留点聚类过程中,使用基于密度的聚类算法对停留点进行聚类,之所以使用基于密度的聚类算法,是因为相比其他聚类算法,该方法有助于检测不规则形状的聚类^[5]。经过聚类分析之后,可以将所有用户的轨迹转换为用停留区域表示的轨迹,该轨迹中的轨迹点就是聚类后的停留区域,这样就可以通过用户的轨迹历史分析用户之间的相似性了,具体的相似性计算方法将在下文给出。

2 相似度计算

2.1 用户轨迹相似度

通过聚类分析得到了用户的历史轨迹,用户经

历过的地点在计算用户兴趣相似度所占的权重与它们在历史轨迹所出现的次数成反比,出现次数越多的地方对于贡献用户兴趣相似度所占的权重越小,反之,出现次数越少的地方对于贡献用户相似度所占的权重越大。借鉴 IR 系统中的逆文档频率,定义了逆轨迹频率,用以反映每个地点对于用户兴趣的权重。

定义 1 逆轨迹频率 (ITF)。逆轨迹频率是轨迹中轨迹点的普遍重要性的度量,可以由总用户数除以访问过该轨迹点的用户的数目,再将得到的商取对数,则轨迹点 r 的逆轨迹频率为

$$\text{ITF}_r = \log \frac{|U|}{|\{u: r \in T_u\}|} \quad (1)$$

式中: r 表示轨迹点, $|U|$ 为用户总数, T_u 表示用户 u 的所有轨迹点的集合, $|\{u: r \in T_u\}|$ 为轨迹中包含轨迹点 r 的用户数目。

为了计算所有用户的轨迹历史的信息指纹,本文首先采用 Geohash 算法将用户访问过的轨迹点编码成 n 位(n 的设定需要综合考虑存储成本以及数据集的大小) 二进制数字,对用户的历史轨迹进行编码,编码后用户 u 的历史轨迹 $s_{u,i}$ 中轨迹点 r 可以表示为 $c_r = \langle c_r^1, \dots, c_r^k, \dots, c_r^n \rangle$ 。为了计算用户之间的相似性,将用户的相似哈希^[7] 编码设置为与停留区域等长的编码,每位编码都初始为 0,那么用户 u 的初始哈希编码 c_u 可以表示为 $\langle c_u^1, \dots, c_u^k, \dots, c_u^n \rangle$,其中每位的编码都为 0,即: $c_u^1 = \dots = c_u^k = \dots = c_u^n = 0$ 。

权重映射规则 $f(c_r^k)$: 如果用户 u 的轨迹 i 中轨迹点 r 的第 k 位编码 c_r^k 对应的二进制数字为 1,则权重为正,否则,权重为负,即

$$f(c_r^k) = \begin{cases} 1, & c_r^k = 1 \\ -1, & c_r^k = 0 \end{cases} \quad (2)$$

定义 2 用户历史轨迹权重。用户历史轨迹权重定义为用户轨迹中所有轨迹点的逆轨迹频率按照权重映射规则按位相加得到的用户历史轨迹的编码序列。用户 u 的第 i 条轨迹的用户历史轨迹权重可以表示为 $w_{u,i}$, 即

$$w_{u,i} = \langle w_{u,i}^1, \dots, w_{u,i}^k, \dots, w_{u,i}^n \rangle \quad (3)$$

$$w_{u,i}^k = c_u^k + \sum_{j=1}^{|s_{u,i}|} f(c_{u,i}^k) \times \text{ITF}_r \quad (4)$$

式中: $|s_{u,i}|$ 表示用户 u 的第 i 条轨迹中轨迹点的个数, $w_{u,i}^k$ 表示用户 u 的第 i 条轨迹的用户历史轨迹权重中的第 k 位编码。通过上式就可得到用户轨迹的加权编码,需要将该编码使用权重编码逆映射规则 $g(w_{u,i}^k)$ 转化为二进制编码。

用户 u 的第 i 条轨迹的第 k 位编码的权重编码逆映射规则为 $g(w_{u,i}^k)$, 根据用户历史轨迹权重进

行映射,如果对应位权重为正数,则映射为 1; 否则,映射为 0:

$$g(w_{u,i}^k) = \begin{cases} 1, & w_{u,i}^k > 0 \\ 0, & w_{u,i}^k \leq 0 \end{cases} \quad (5)$$

通过以上的权重编码逆映射规则,可以将用户历史轨迹权重转换为二进制编码,即用户 u 的第 i 条轨迹的兴趣指纹为 $F_{u,i}$:

$$F_{u,i} = \langle g(w_{u,i}^1), \dots, g(w_{u,i}^k), \dots, g(w_{u,i}^n) \rangle \quad (6)$$

得到用户的兴趣指纹之后,使用海明距离表示用户之间的差异性。海明距离指兴趣指纹的对应比特取值不同的比特数,该值越小表示兴趣指纹的差异性越小。将海明距离归一化处理后,可以得到轨迹 x 和 y 之间的差异率 $d_{x,y}$:

$$d_{x,y} = \frac{H(F_{u,x}, F_{v,y})}{n} \quad (7)$$

式中: $H(F_{u,x}, F_{v,y})$ 表示轨迹的海明距离。通过差异率,可以计算轨迹 x 和 y 之间的相似性如下:

$$\text{SIM}_{x,y} = 1 - d_{x,y} \quad (8)$$

用户历史轨迹的相似性通过计算用户的所有轨迹之间的相似性获得。用户 u 和用户 v 的轨迹相似性为

$$\text{RSIM}_{u,v} = \frac{\sum_{p=1}^{|T_u|} \sum_{q=1}^{|T_v|} \text{SIM}_{x,y}}{|T_u| \times |T_v|} \quad (9)$$

式中: $|T_u|$ 和 $|T_v|$ 分别表示用户 u 和用户 v 的轨迹历史中轨迹的数目。如果用户 u 和用户 v 的轨迹相似性越大,说明 2 个用户访问的地理轨迹越相似,那么 2 个用户的兴趣习惯也越相似。

2.2 地点相似度

在为用户推荐地点的时候,不仅要考虑用户轨迹之间的相似性,还要考虑地点之间的相似性。本文在计算相似度时,不仅仅考虑用户兴趣相似度,而且也考虑地点之间的相似度。在计算地点相似度时,本文借鉴基于物品的协同过滤的思想对地点相似度进行计算。常见的计算物品相似性的方法有余弦相似度^[10] 和条件概率^[11] 等。条件概率的计算公式为

$$\text{CPIS}_{i,j} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)|} \sqrt{|N(j)|}} \quad (10)$$

分析式(10)可以发现,如果物品 i 是非常热门的物品,那么式(10)的分子 $|N(i) \cap N(j)|$ 就会相当于 $|N(j)|$ 。虽然式(10)已经对物品 i 的流行度进行了考虑,但是热门的物品 i 仍然会获得很高的相似度。缓解这一情况的方法^[12-13] 如下

$$\text{PPLS}_{i,j} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)|}^\lambda \sqrt{|N(j)|}} \quad (11)$$

式(11)通过提高 λ 的值来缓解热门物品 i 对于

相似度计算的影响。但是该公式存在一定的局限性,其中 λ 的取值只能取一个固定值,并不能很好的区别不同物品的流行度,所以推荐效果有待提高。本文在计算地点相似度时,充分考虑地点流行度这一因素,并对相似度进行归一化,以提高推荐的准确度。 λ 的取值采用动态的值,该值由2个地点的流行度决定。 λ 取值为地点 i 的流行度与地点 i 、地点 j 流行度之和的比值。即动态惩罚地点流行度的相似度(dynamic penalized popular location similarity, DP-PLS)计算方法如下:

$$DPPLS_{ij} = \frac{|N(i) \cap N(j)|}{\sqrt{|N(i)|}^\lambda \sqrt{|N(j)|}^{1-\lambda}} \quad (12)$$

$$\lambda = \frac{|N_i|}{|N_i| + |N_j|} \quad (13)$$

经过上面的改进就可以对不同地点的流行度进行考虑,动态缓解热门地点对于地点相似度计算的影响,相对于静态的缓解热门地点对于地点相似度的计算的影响更加有效,可以更好地提高推荐的准确度。

3 混合轨迹和地点相似性的推荐算法

在传统的推荐系统中,一般基于用户对物品的评分,向用户推荐物品。评分一般分为两种:显示评分和隐式评分。显示评分指用户对物品直接给出的喜欢和不喜欢或者具体的分值等评价,而隐式评分一般包括评语、在线浏览时间和购买次数等。基于GPS轨迹无法获取用户对每个地点的显示评分,因此只能使用隐式评分评估用户对地点的兴趣。在这里使用用户对每个地点的访问次数作为用户对某个地点的隐式评分。为了对不同用户的访问次数进行统一比较,需要将每个用户的访问次数进行归一化处理。归一化处理使用如下的公式计算:

$$R_{ur} = \frac{V_{ur}}{\sum_j |T_u| V_{uj}^2} \quad (14)$$

式中: V_{ur} 表示用户 u 访问轨迹点 r 的次数, $|T_u|$ 表示用户访问过的所有轨迹点的总数。

已有的推荐算法仅仅考虑用户相似度,或者仅仅考虑物品的相似度。本文将同时考虑用户相似度和位置相似度,从而为用户推荐更可能感兴趣的地点。这里需要使用参数 α 和 β 决定2个相似性对推荐的影响权重,要求 $\alpha + \beta = 1$ 。用户 u 对轨迹点 tr 的潜在兴趣可以表示为

$$R_{ur} = \alpha \times \frac{\sum_{v \in U, v \neq u} RSIM_{uv} \times R_{vr}}{\sum_{v \in U, v \neq u} RSIM_{uv}} +$$

$$\beta \times \frac{\sum_{r_i \in T_u, r_i \neq r} DPPLS_{r_i r} \times R_{ur}^{r_i}}{\sum_{r_i \in T_u, r_i \neq r} DPPLS_{r_i r}} \quad (15)$$

式中: T_u 表示用户 u 所有访问过的地点, U_r 表示访问过轨迹点 r 的用户的集合。这样便可以获得用户对未访问过的地点的潜在兴趣,从而可以向用户推荐潜在兴趣较大的Top-N地点,进一步可以将推荐地点附近的饭店、商场等信息推荐给用户。

4 对比试验和结果分析

4.1 数据集介绍

使用的数据集是微软亚洲研究院提供的GPS数据集^[14-15],其中包含用户在2007年-2012年间的轨迹,在该数据集中每条GPS轨迹都是一系列的坐标点,其中每个坐标点都是由经度、维度和时间3种信息组成。本文将该数据集按照时序顺序进行了划分,将前80%的数据用作训练集,将余下的20%的数据用作测试集,同时为了避免过耦合问题,本文进行了5次交叉确认实验。

4.2 实验评价指标

实验采用召回率和准确率作为评价标准。准确率和召回率是信息检索和自然语言处理中经常使用的评价指标,也是评测推荐算法精度的常用指标。准确率(precision)和召回率(recall)分别表示推荐产品中用户真实购买产品的比例和用户真实购买产品中属于推荐产品的比例:

$$\text{precision} = \frac{1}{|U|} \times \sum_{u \in U} \frac{|R_u \cap T_u|}{|T_u|} \quad (16)$$

$$\text{recall} = \frac{1}{|U|} \times \sum_{u \in U} \frac{|R_u \cap T_u|}{|R_u|} \quad (17)$$

式中: R_u 表示向用户 u 推荐的地点集合, T_u 表示在测试集中用户 u 真实访问过的地点集合。

覆盖率描述推荐系统发掘长尾物品的能力,一般指推荐系统所推荐的地点占系统中所有地点的比例。该比例越高,表明推荐系统的覆盖率越高,长尾地点被用户访问的概率也越大。覆盖率(coverage)计算公式如下

$$\text{coverage} = \frac{1}{|L|} \times \left| \bigcup_{u \in U} R_u \right| \quad (18)$$

式中: R_u 表示向用户 u 推荐的地点集合, L 表示系统中所有的地点集合。

4.3 实验参数的设定

在实验中,提取停留点算法有2个重要的参数:时间阈值和距离阈值。如果时间阈值和距离阈值选取的太小,则会生成很多的停留点,这些停留点中存在很多无意义的噪音数据。如果时间阈值和距离阈值太大,停留点就会太少,精度就会减小。

根据文献[5]设定时间阈值为30 min,停留点距离阈值为200 m,在聚类分析时,DBSCAN算法半径设定为500 m,簇内最少停留点个数设定为2。

4.4 实验结果及分析

将本文的动态惩罚地点流行度的相似度方法(DPPLS)与余弦相似度方法(CS)^[10]和惩罚地点流行度的相似度方法(PPLS)^[12-13]进行对比。该组实验是为了验证动态惩罚地点流行度的相似度方法在计算地点的相似上是否比传统的余弦相似度方法、惩罚地点流行度的相似度方法表现出色。为了对比3种方法的效果,将相似度用于地点推荐中,根据推荐算法的召回率、准确率和覆盖率进行比较。具有较高推荐精确度并且能够覆盖较多地点的相似度方法被认为是较好的相似度分析算法(具体结果如图1~3)。

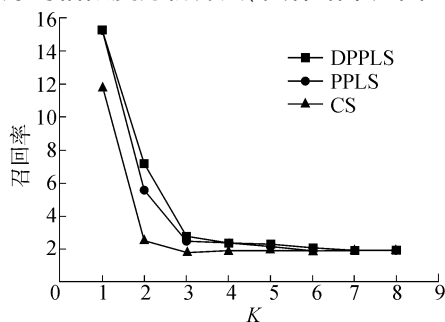


图1 DPPLS、PPLS 和 CS 的召回率对比

Fig. 1 Comparison of recall for DPPLS, PPLS and CS

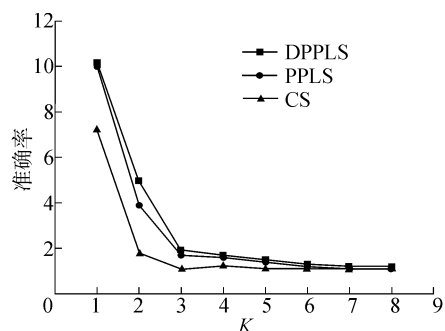


图2 DPPLS、PPLS 和 CS 的准确率对比

Fig. 2 Comparison of precision for DPPLS, PPLS and CS

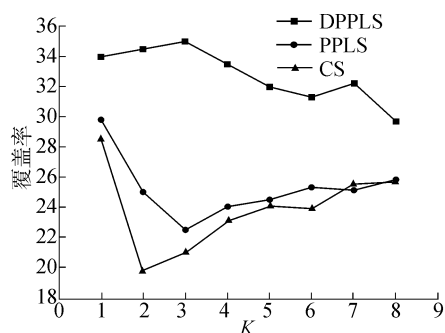


图3 DPPLS、PPLS 和 CS 的覆盖率对比

Fig. 3 Comparison of coverage for DPPLS, PPLS and CS

从图1可以看到,CS算法和PPLS算法比DPPLS算法在召回率方面要低,当 $K \geq 6$ 时三者召回率趋于同一个值。

从图2可以看到,当 $K < 6$ 时,CS算法和PPLS算法比DPPLS算法在准确率低,当 $K \geq 6$ 时,三者准确率趋于同一个值。

从图3可以看到,CS算法和PPLS算法的覆盖率要比DPPLS算法低很多,表明本文提出的DPPLS算法要比CS算法和PPLS算法在推荐长尾地点的能力强。

以下对比的是基于用户的协同过滤(UserCF)、基于位置的协同过滤(LocationCF)和基于用户和位置的混合协同过滤(ULCF)在召回率和准确率方面的对比。

如图4和5所示,推荐的召回率和准确率并不和参数 K 成线性关系。UserCF算法、LocationCF算法和ULCF算法都在随着 K 值的增大呈现逐渐减小的趋势。本文提出的ULCF算法在召回率和准确率方面都要高出UserCF算法和LocationCF算法。

如图6所示,LocationCF算法随着 K 值的变大覆盖率先变小后变大。UserCF和ULCF算法都在随着 K 值变大覆盖率先变大后变小,并且ULCF算法的覆盖率在 K 大于4后显著强于UserCF和LocationCF算法。

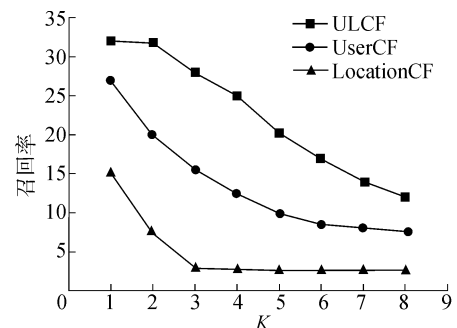


图4 ULCF、UserCF 和 LocationCF 的召回率对比

Fig. 4 Comparison of recall for ULCF, UserCF and LocationCF

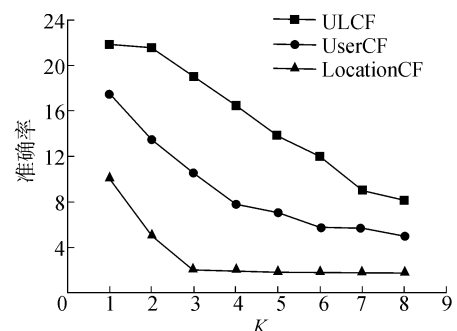


图5 ULCF、UserCF 和 LocationCF 的准确率对比

Fig. 5 Comparison of precision for ULCF, UserCF and LocationCF

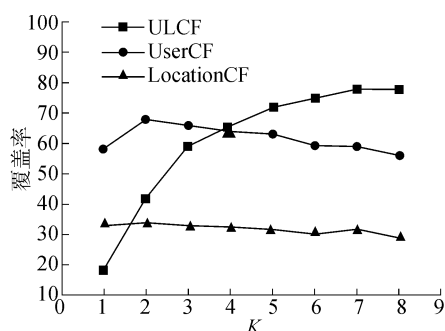


图6 ULCF、UserCF 和 LocationCF 的覆盖率对比

Fig. 6 Comparison of coverage for ULCF, UserCF and LocationCF

综上,ULCF 算法在 K 值为 3 时,召回率、准确率和覆盖率都取得相对较好的性能。通过实验表明,使用相似哈希计算用户的轨迹相似性是可行的,并且通过混合轨迹相似性和地点相似性可以为用户提供比较好的推荐服务。

5 结论

本文针对传统的基于时空轨迹数据分析用户的相似性的方法中存在的问题,研究了使用相似哈希分析用户轨迹的相似性的方法,同时考虑地点流行度对地点相似性的影响,提出了混合轨迹相似性和动态惩罚流行度的地点相似性的方法,根据该相似性提供个性化的地点推荐服务,得到如下结论:

- 1) 使用相似哈希的方法能够加快轨迹数据的分析,提高轨迹计算的效率;
- 2) 考虑地点的流行度动态惩罚热门地点在地点相似性计算中的权重能够显著提高地点推荐的准确度;
- 3) 将轨迹的相似性与地点的相似性相结合,能够比单独使用轨迹相似性和地点相似性取得更好的推荐效果。

本文进一步的研究工作如下: 用户访问某个地点,不只取决于自己的兴趣,还受用户好友的影响,需要分析用户的好友关系对用户兴趣的影响权重。

参考文献:

- [1] 马宇驰,杨宁,谢琳,等. 基于轨迹时空关联语义和时态熵的移动对象社会角色发现[J]. 计算机研究与发展, 2012, 49(10): 2153-2160.
MA Yuchi, YANG Ning, XIE Lin, et al. Social roles discovery of moving objects based on spatial-temporal associated semantics and temporal entropy of trajectories[J]. Journal of computer research and development, 2012, 49(10): 2153-2160.
- [2] LI Mu, AHMED A, SMOLA A J. Inferring movement trajectories from GPS snippets[C]//Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. New York, USA, 2015: 325-334.

- [3] ZHONG Haidong, ZHANG Shaozhong, WANG Yanling. Mining users' similarity from moving trajectories for mobile e-commerce recommendation[J]. International journal of hybrid information technology, 2014, 7(4): 309-320.
- [4] 李晓静,张晓滨. 基于 LCS 的用户时空行为兴趣相似性计算方法[J]. 计算机工程与应用, 2013, 49(20): 251-254.
LI Xiaojing, ZHANG Xiaobin. Computing user similarity of spatio-temporal behaviour and interests based on LCS[J]. Computer engineering and applications, 2013, 49(20): 251-254.
- [5] LI Quannan, ZHENG Yu, XIE Xing, et al. Mining user similarity based on location history[C]//Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. New York, USA, 2008.
- [6] YING J J C, LU E H C, LEE W C, et al. Mining user similarity from semantic trajectories[C]//Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks. New York, USA, 2010: 19-26.
- [7] CHARIKAR M S. Similarity estimation techniques from rounding algorithms[C]//Proceedings of the thirty-fourth Annual ACM Symposium on Theory of Computing. New York, USA, 2002: 380-388.
- [8] SADOWSKI C, LEVIN G. Simhash: Hash-based similarity detection[R]. Technical Report, Google, 2007: 1-10.
- [9] ZHENG Yu, ZHOU Xiaofang. Computing with spatial trajectories[M]. New York, USA: Springer, 2011: 243-325.
- [10] ADOMAVICIUS G, TUZILIN A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions[J]. IEEE transactions on knowledge and data engineering, 2005, 17(6): 734-749.
- [11] KITTS B, FREED D, VRIEZE M. Cross-sell: a fast promotion-tunable customer-item recommendation method based on conditionally independent probabilities[C]//Proceedings of the sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2000: 437-446.
- [12] DESHPANDE M, KARYPIS G. Item-based top-N recommendation algorithms[J]. ACM transactions on information systems, 2004, 22(1): 143-177.
- [13] 邢春晓,高凤荣,战思南. 适应用户兴趣变化的协同过滤推荐算法[J]. 计算机研究与发展, 2007, 44(2): 296-301.
XING Chunxiao, GAO Fengrong, ZHAN Si'nan. A collaborative filtering recommendation algorithm incorporated with user interest change[J]. Journal of computer research and development, 2007, 44(2): 296-301.
- [14] ZHENG Yu, ZHANG Lizhu, XIE Xing, et al. Mining interesting locations and travel sequences from GPS trajectories[C]//Proceedings of the 18th international conference on World wide web. New York, USA, 2009: 791-800.
- [15] ZHEGN Yu, LI Quannan, CHEN Yukun, et al. Understanding mobility based on GPS data[C]//Proceedings of the 10th International Conference on Ubiquitous Computing. New York, USA, 2008: 312-321.