第 45 卷 第 1 期 计 算 机 工 程 2019 年 1 月
Vol. 45 No. 1 Computer Engineering January 2019

·安全技术· 文章编号:1000-3428(2019)01-0096-07 文献标志码:A 中图分类号:TP391.3

基于 LSH 的隐私保护 POI 推荐算法

沈鑫娣[1],翟东君[1],张得天[2],刘 安[1]

(1. 苏州大学 计算机科学与技术学院,江苏 苏州 215006;
2. 江南大学 数字媒体学院,江苏 无锡 214122)

摘 要: 基于位置的社交网络利用用户的签到数据进行兴趣点(POI)推荐,但是出于对数据隐私的考虑,各种社交平台之间不愿意直接共享数据。为综合各个社交平台的数据从而提供更好的 POI 推荐服务,提出一种基于局部敏感哈希(LSH)的隐私保护 POI 推荐算法。通过 LSH 选取相似用户集合,极大地减少计算量,满足用户的快速响应需求。利用 LSH 和 Paillier 同态加密技术,在计算过程中保护数据隐私不被泄露。真实数据集上的实验结果表明,在响应时间和预测准确度上,该算法优于传统基于用户的协同过滤推荐算法。

关键词: 局部敏感哈希; 隐私保护; 推荐算法; 兴趣点; 同态加密

中文引用格式: 沈鑫娣,翟东君,张得天,等. 基于 LSH 的隐私保护 POI 推荐算法[J]. 计算机工程,2019,45(1):96-102.
英文引用格式: SHEN Xindi, ZHAI Dongjun, ZHANG Detian, et al. Privacy preserving POI recommendation algorithm based on LSH[J]. Computer Engineering, 2019, 45(1): 96-102.

Privacy Preserving POI Recommendation Algorithm Based on LSH

SHEN Xindi[1], ZHAI Dongjun[1], ZHANG Detian[2], LIU An[1]

(1. School of Computer Science and Technology, Soochow University, Suzhou, Jiangsu 215006, China;
2. School of Digital Media, Jiangnan University, Wuxi, Jiangsu 214122, China)

【Abstract】 The Location-Based Social Network (LBSN) uses the user's check-in data to recommend the Point of Interest (POI), but for the consideration of data privacy, various social platforms are unwilling to share data directly. In order to provide a better POI recommendation service by synthesizing the data of various social platforms, a privacy preserving POI recommendation algorithm based on Locality-Sensitive Hashing (LSH) is proposed. The similar user set is selected by LSH, which greatly reduces the computation cost and satisfies the user's rapid response demand. LSH and Paillier homomorphic encryption techniques are used to protect data privacy from disclosure. Experimental results on real data sets show that the proposed algorithm is superior to the traditional collaborative filtering recommendation algorithm based on users in response time and prediction accuracy.

【Key words】 Locality-Sensitive Hashing (LSH); privacy preserving; recommendation algorithm; Point of Interest (POI); homomorphic encryption

DOI: 10.19678/j.issn.1000-3428.0049731

0 概述

随着带有 GPS 定位系统的移动设备和无线通信技术的快速发展,基于位置的服务(Location-Based Services, LBS)受到越来越多的关注,很多用户在社交应用上分享他们的信息。在基于位置的社交网络(Location-Based Social Network, LBSN)中,利用用户的签到数据进行兴趣点(Point of Interest, POI)推荐成为一个重要的研究点。签到数据反映了用户对地点场所的偏好,从而为个性化的 POI 推荐提供坚实的基础。这种推荐方式不仅可以让用户在不花费太多时间的情况下搜索到新的相关地点,而且可以让服务提供商对用户提供精确的推荐服务。

虽然推荐系统在过去几十年中已经得到较多研究[1],但是个性化 POI 推荐[2-3]因为其特点最近才受到大家的关注。与传统推荐系统中的用户-项目评分矩阵相比,POI 推荐中的用户-签到矩阵通常更加稀疏。例如,Gowalla 数据集的密度是 2.08×10^{-4},而 Netflix 的数据集密度是 0.01。用户签到数据的稀疏性使得推荐系统难以捕捉用户对地点的偏好。

基金项目: 国家自然科学基金面上项目(61572336);国家自然科学基金青年基金(61702227)。
作者简介: 沈鑫娣(1992—),女,硕士研究生,主研方向为时空数据、数据隐私;翟东君,硕士研究生;张得天,讲师;刘 安,副教授。
收稿日期: 2017-12-18 **修回日期:** 2018-01-18 **E-mail:** 20154227008@stu.suda.edu.cn

地理位置的影响是区分 POI 推荐和传统物品推荐的另一个因素。根据签到数据的分析结果显示,用户通常喜欢从近的地方向远的地方移动。此外,用户的历史签到信息通常是本地密集的,这使得冷启动问题在 POI 推荐中更为突出,原因是即使用户在自己的住宅区附近访问了足够多的地方,当用户旅行到一些新的地区时也不可避免地会遇到冷启动问题。

为解决矩阵稀疏和冷启动问题,可以通过收集多个分布式平台的数据来做 POI 推荐,比如,用户 A 在微博上有签到数据,用户 B 在 Facebook 上有签到数据,把微博和 Facebook 上的用户签到数据结合起来做 POI 推荐。但是在这种情况下,也存在两大挑战。一方面,由于数据隐私问题,微博和 Facebook 都不愿意将自己内部的用户签到数据向对方公开,这使得计算用户 A 和用户 B 之间的相似度之后再做进一步推荐变得困难。另一方面,用户签到数据分布在多个平台,平台之间不可避免地需要消息通信,导致消耗很多时间,不能满足用户的快速响应需求。

基于以上 2 个挑战,本文提出一种基于局部敏感哈希(Locality-Sensitive Hashing,LSH)隐私保护的 POI 推荐算法。该算法研究了分布式平台下的 POI 推荐问题,采用 LSH 和同态加密技术相结合的方式辅助 POI 推荐,保障推荐的准确性、隐私性和高效性,并通过在真实数据集上的实验来验证其可行性。

1　相关工作

协同过滤是目前推荐系统使用比较广泛的技术[4]。协同过滤分为 2 类:基于记忆(Memory-Based)的协同过滤和基于模型(Model-Based)的协同过滤[5]。

基于记忆的协同过滤又可以分为基于用户和基于项的协同过滤。对于基于用户的系统,所有用户之间的相似度是根据他们对相关项目的评分,利用相似性度量方法计算得到,然后通过相似用户对同一项目的评分加权得到对缺失项的评分。对于基于项的系统,则是找到类似评分的项,利用相似项的用户评分来进行预测。

基于模型的协同过滤是利用用户的历史评分数据训练出一个模型,然后根据这个模型进行预测,常见的基于模型的协同过滤有矩阵分解、基于聚类的方法等。这些传统的协同过滤方法都不适用分布式系统下的数据推荐,因此本文采用 LSH 方法辅助 POI 推荐。

目前有很多研究关注用户数据隐私。本文也考虑了平台用户数据隐私的问题。同态加密[6]是一种常见的加密方法[7-9]。同态加密允许在密文上计算以达到隐私保护的目的。文献[10]提出一个利用 Yao 电路[11]和同态加密来保护用户隐私的协同过滤推荐框架,既保护了用户的隐私同时又保证了推荐结果的准确性。文献[12-13]通过利用同态加密和 Yao 电路技术提出基于模型的 2 种推荐算法:矩阵分解和岭回归的解决方案。文献[14]提出一个结合同态加密和数据打包技术的隐私保护推荐方法。虽然这些方案通过安全的多方计算协议来提供隐私保护,但是同态加密需要消耗大量的计算时间和通信代价,不能满足用户的快速响应需求,因此使用同态加密把分布式系统上用户数据全部加密的方法不能满足现代社交应用的实际需求。

文献[15]提出 k-匿名方法,通过概括和隐匿技术,发布精度较低的数据,在 k-匿名的数据集中,每条记录都至少和 $k-1$ 条记录具有完全相同的属性值。但是这种方法在敏感数据方面缺乏多样性或者攻击者有辅助信息的情况下面临隐私暴露的风险[16]。解决隐私暴露问题的另一种方法是随机干扰技术。文献[17]表示在原始用户数据中加入特定分布的随机噪声防止信息泄露之后,仍能得到准确的推荐结果。最近有一项工作就是使用随机干扰作为数据混淆技术来形成一个简单而有效的隐私保护框架[18]。然而,随机噪声的范围一般是根据经验选择,并没有可证明的隐私保证。另外,文献[19]指出通过在扰动数据上应用聚类方法,攻击者在很大程度上可以推断出用户的私人数据,从而造成隐私泄露。

差分隐私(Differential Privacy,DP)[20-21]是一种严格可证明的隐私定义。差分隐私可以无视攻击者的背景知识,保护个人的信息不能通过基于整个数据集的计算结果推断出来。目前也有很多工作利用差分隐私保护数据的安全性。文献[22]将差分隐私应用到非社会化推荐。然而其不能适用于社会化推荐,为克服这个缺点,文献[23]提出在根据社交网络结构对用户进行分组的聚类之后,再通过差分隐私保护用户数据安全。文献[24]提出一个基于距离的差分隐私框架作为差分隐私的扩展应用。文献[25]通过结合随机干扰和差分隐私,提出一种混合的隐私保护推荐系统,用户数据的隐私由随机干扰保护,而推荐结果的隐私由差分隐私保护。文献[26]提出一种在服务器不受信任的情况下,在客户端利用差分隐私保护用户隐私的实用方法。

2　问题定义

假设目标用户 u_1 是一个微博用户,在地点 $\{l_1, l_2, \cdots, l_{n_1}\}$ 有签到数据;用户 u_2 是一个 Facebook 用户,在地点 $\{l_1, l_2, \cdots, l_{n_2}\}$ 有签到数据;用户 u_3 是一个 Gowalla 用户,在地点 $\{l_1, l_2, \cdots, l_{n_3}\}$ 有签到数据。当

使用基于用户的协同过滤方法时,第一步就是计算目标用户与其他用户之间的相似度,但是现在数据分布在多个平台上,因此会产生以下问题:1)考虑到各个平台的隐私问题,Facebook 和 Gowalla 平台上的数据不会直接发送给微博平台;2)各个平台上的数据量可能都非常庞大,那么计算相似度的时间和通信量也会非常大,不能满足目标用户的快速响应需求。

为解决上述问题,本文提出一种基于 LSH 的隐私保护 POI 推荐算法,可以很好地处理分布式平台下的推荐问题。

为更好地阐述该方法,首先定义一些标识符。

1)$PF=\{pf_1,pf_2,\cdots,pf_k\}$ 表示 k 个分布式平台的集合。

2)$U=\{U_1,U_2,\cdots,U_k\}$ 表示与 k 个分布式平台相对应的用户集合。对于平台 pf_k,其用户集合表示如下:$U_k=\{u_{k_1},u_{k_2},\cdots,u_{k_m}\}$,其中 m 是用户总数。

3)$L=\{l_1,l_2,\cdots,l_n\}$ 表示 n 个签到地点,为方便下文讨论,假设所有平台的签到地点都是一样的,即 $n_1=n_2=n_3=n$。

3 本文算法

3.1 LSH

LSH[27]在被提出之后就广泛应用于分布式系统中。LSH 的基本思想是将 2 个点冲突的可能性与其距离紧密相连,即 2 个点距离越近,它们冲突的可能性越高,2 个点距离越远,它们冲突的可能性则越低[28]。

定义 1(局部敏感哈希) LSH 依赖于一个哈希函数族,该函数族是空间 R^d 中点域 S 到某个集合域 D 的一组映射函数,表示为 $H=\{h:S\rightarrow D\}$。随机选择一个哈希函数 $h_i(i=1,2,\cdots,N)$,对 R^d 内的数据点 a 和 b 进行哈希,若满足以下条件:

1)若 $||a-b||\leqslant r_1$,则 $p[h(a)=h(b)]\geqslant p_1$。

2)若 $||a-b||\geqslant r_2$,则 $p[h(a)=h(b)]\leqslant p_2$。

其中,$p[\cdot]$是概率函数,$0<r_1\leqslant r_2$,$0\leqslant p_2<p_1\leqslant 1$,则称其为$(r_1,r_2,p_1,p_2)$-位置敏感哈希函数族。

如图 1 所示,原始 L 个数据点通过哈希函数 $h(\cdot)$映射到 t 个哈希桶中,每个桶包含的数据点是相邻的数据点,其数目 $|b_i|$ 远远小于 L,同时也达到数据降维的目的。如果一个目标用户想要查询与数据点 X 相似的数据点,那么只要通过哈希函数 $h(X)$映射到对应的桶中,桶中的数据点就有极大概率是该目标用户的相似数据点。由于桶中数据点的个数远远小于 L,因此极大地提高了搜索的效率。除此之外,通过哈希映射这个步骤,也可以保障用户原始数据的隐私,即用户只知道哈希值但是不知道其他用户的原始数据。因此,LSH 在分布式平台中既可以提供高效的查询,也可以提供有效的隐私保护。

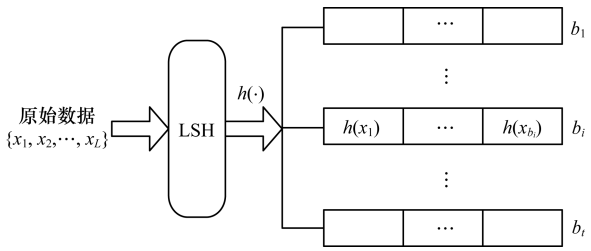

图 1 LSH 原理

3.2 Paillier 同态加密

Paillier 是基于合数分解的困难性概率公钥密码系统,Paillier 加密系统是一个加法同态的加密体制,本文利用 Paillier 保证在计算过程中不泄露数据信息。对 Paillier 密码体制的同态性分析[29]如下:

若 Paillier 的公私钥为:公钥(N,g),私钥(λ,μ),那么对于明文 $m(m\in Z_n)$,选择随机数 $r(r\in Z_n^*)$,加密过程为:$c\equiv g^m\cdot r^N \bmod N^2$,解密过程为:$m\equiv L(c^\lambda \bmod N^2)\cdot\mu \bmod N$。对于明文 m_1 和 m_2,加密后有:$E(m_1)\equiv g^{m_1}x_1^N(\bmod N^2)$,$E(m_2)\equiv g^{m_2}x_2^N(\bmod N^2)$。由此可得:

$$E(m_1)\cdot E(m_2)\equiv g^{m_1}x_1^N\cdot g^{m_2}x_2^N \bmod N^2\equiv$$
$$g^{m_1+m_2}(x_1x_2)^N \bmod N^2\equiv E(m_1+m_2)$$

因此,Paillier 公钥密码体制满足加法同态特性。

3.3 算法设计

基于 LSH 的隐私保护 POI 推荐算法包括 3 个部分:

1)离线用户索引建立,选择 LSH 函数族,根据历史用户签到数据,将分布式平台上的用户映射到对应的桶中。

2)在线相似用户查找,根据选择的 LSH 函数族,目标用户会映射到一个桶中,该桶中其他用户在很大概率上被认为是目标用户的相似用户。

3)目标用户地点推荐,根据上一步得到的相似用户集合,利用相似用户的签到数据信息预测目标用户对某些未签到地点的喜好程度,由此得到推荐结果。

3.3.1 离线用户索引建立

首先选择一个 LSH 函数 $h(\boldsymbol{u})$或者一个 LSH 函数族 $H=\{h_1(\boldsymbol{u}),h_1(\boldsymbol{u}),\cdots,h_r(\boldsymbol{u})\}$为分布在不同平台上的用户建立索引。哈希函数的选择基于定义 1 的距离定义,因为皮尔逊相关系数在推荐系统中经常被作为计算相似度或者衡量距离的方法,所以本文采用皮尔逊相关系数对应的 LSH 函数进行索引建立。

对于一个用户 u,其历史签到数据可以表示为一个 n 维的向量 $\boldsymbol{u}=(ch_{l_1},ch_{l_2},\cdots,ch_{l_n})$,其中 ch_{l_i}代表用户的签到情况,如果 $ch_{l_i}=0$ 表明用户没有访问过

l_i这个地点。$\boldsymbol{v}=(v_1,v_2,\cdots,v_n)$是一个 n 维的向量,其中 v_i是在[-1,1]范围里面选取的随机数,符号。代表向量之间的点乘操作。哈希函数定义如下:

$$h(\boldsymbol{u})=\begin{cases}1,\boldsymbol{u}\circ\boldsymbol{v}>0\\0,\boldsymbol{u}\circ\boldsymbol{v}\leqslant 0\end{cases}$$

通过哈希函数,用户 u 被散列成 0 或者 1 的二进制值。LSH 本质上是基于概率的方法,因此引入更多的哈希函数或哈希表,可以获得更准确的相似度。本文算法设计就是假设有 T 个 LSH 表,每个表都由 r 个 LSH 函数组成。在每个 LSH 表中,每个用户 u 在 LSH 之后都有一个对应的 r 维二进制哈希向量:$\boldsymbol{H}(\boldsymbol{u})=(h_1(\boldsymbol{u}),h_2(\boldsymbol{u}),\cdots,h_r(\boldsymbol{u}))$。$\boldsymbol{H}(\boldsymbol{u})$的每一项值为 0 或者 1。如果在 T 个哈希表中,存在一个哈希表,令 u_1 和 u_2 在 LSH 之后被散列到相同的桶中,则 u_1 和 u_2 就是相似的邻居。利用这种方法,就可以为不同平台上的用户建立索引。为满足用户快速响应的需求,提高查找效率,不同平台的索引存在服务提供商端。根据上文分析,利用 LSH 建立的索引并不会暴露平台用户的签到数据,因此没有隐私泄露的问题。

3.3.2　在线相似用户查找

在基于 LSH 建立的索引基础上,目标用户想要查询相似用户可以根据本人的签到记录利用 LSH 哈希函数计算散列值,然后在服务提供商存储的索引中根据该散列值找到相应的相似用户,该散列值对应的桶中所有用户都被认为是相似的用户。

3.3.3　目标用户地点推荐

本节将介绍利用相似用户给目标用户推荐的方法。为保护分布式平台上用户的数据隐私,用户的地点签到数据用 Paillier 加密之后跟散列值存储在一起。对于目标用户想要预测的地点 l_{tar},可以计算如下:

$$ch_{l_{\text{tar}}}=\frac{1}{|SIM|}D\left(E\left(\sum_{u_i\in SIM}ch_{l_{\text{tar}i}}\right)\right)=\frac{1}{|SIM|}D\left(\prod_{u_i\in SIM}E\left(ch_{l_{\text{tar}i}}\right)\right)$$

其中,SIM 代表相似用户的集合,$|SIM|$代表相似用户的个数,$ch_{l_{\text{tar}i}}$代表用户 u_i在地点 l_{tar}的签到数据,$E(ch_{l_{\text{tar}i}})$代表 Paillier 加密之后的用户 u_i在地点 l_{tar}的签到数据,$D(\cdot)$代表 Paillier 解密。根据同态加密的性质,综合考虑相似用户对预测地点的访问次数,$ch_{l_{\text{tar}}}$即为对地点 l_{tar}的预测结果。

基于 LSH 的隐私保护 POI 推荐算法,综合利用不同分布式平台上的用户签到数据为目标用户做 POI 推荐,同时保护各个平台的数据隐私。该算法的伪代码描述如下。

算法 1　基于 LSH 的隐私保护 POI 推荐算法

输入　分布式平台集合 $PF=\{pf_1,pf_2,\cdots,pf_k\}$,各个分布式平台对应的用户集合 $U=\{U_1,U_2,\cdots,U_k\}$,对于平台 pf_k,其用户集合 $U_k=\{u_{k1},u_{k2},\cdots,u_{km}\}$,签到地点集合 $L=\{l_1,l_2,\cdots,l_n\}$,目标用户 u 想要预测的地点 l_{tar}

输出　l_{tar}的预测结果

```
//建立离线用户索引
for t = 1 to T do//T 个哈希表
for i = 1 to k do//k 个分布式平台
for j = 1 to m do
H_t(u_ij) = (h_t1(u_ij), h_t2(u_ij), ⋯, h_tr(u_ij));
for p = 1 to r do
for q = 1 to n do
v_tpq = random[-1,1];
end for
if u_ij ∘ v_tp > 0
then h_tp(u_ij) = 1;
else h_tp(u_ij) = 0;
end if
end for
end for
end for
end for
//相似用户查找
初始化 SIM = ∅; //相似用户集合
for t = 1 to T do
H_t(u) = (h_t1(u), h_t2(u), ⋯, h_tr(u));
for p = 1 to r do
if u ∘ v_tp > 0
then h_tp(u) = 1;
else h_tp(u) = 0;
end if
end for
根据 H_t(u)找到相应的桶号,将相似用户加入 SIM;
end for
//目标用户地点推荐
tmp = E(0);//初始化
for i = 1 to |SIM| do
tmp = tmp · E(ch_l_tar_i);//根据 Paillier 同态加密的性质
end for
ch_l_tar = (1/|SIM|) D(tmp);
return ch_l_tar;
```

4　实验结果与分析

为验证本文算法的执行效率、预测准确度以及不同数据规模对该算法的影响,在真实数据集上对本文算法进行实验。所有算法都是基于 JAVA 1.8 实现,用于实验的是一台 Inter Core(TM) i5-3470 CPU @3.2 GHz 内存 8 GB 64 位 Window7 系统的 PC 机。

4.1　实验数据和度量标准

本文实验数据来源于社交网站 Gowalla(http://snap.stanford.edu/data/loc-gowalla.html),该数据集包含 6 442 890 条签到数据。因为 Gowalla 数据集本身非常稀疏,密度仅为 2.08×10^{-4},所以本文实验抽取其中 10 000 个用户以及对应的 5 000 个地点的签

到数据进行实验,该实验数据集的密度是 2.99×10^{-3},其中,实验数据集中 90% 签到数据作为训练集,10% 签到数据作为测试集。

为衡量 POI 预测的准确性,本文使用均方根误差(Root Mean Square Error,RMSE)作为度量标准。通过计算预测用户签到情况与实际用户签到情况之间的偏差来度量预测的准确性,通常 RMSE 值越小表示预测的准确度越高。假设 qch_i 表示预测用户签到情况,ch_i 表示实际用户签到记录,L_{tar} 表示将要预测的地点集合,$|L_{tar}|$ 表示预测地点集合的数量。RMSE 值的计算公式如下:

$$R_{RMSE}=\sqrt{\frac{\sum_{i\in L_{tar}}(qch_i-ch_i)^2}{|L_{tar}|}}$$

4.2 实验设计与结果分析

将本文算法(下文用 LSH-PR 表示)与传统基于用户的协同过滤推荐算法(下文用 UCF 表示)作对比实验。在实验中,T 代表哈希表的个数,r 代表哈希函数的个数,m 表示用户数目,n 表示地点数目。

4.2.1 不同推荐方法运行效率对比

2 种推荐算法的运行效率对比如图 2、图 3 所示。设置 $T=10$,$r=10$,图 2 中用户数量 m 从 2 000 ~ 10 000 变化,地点数目 $n=5\,000$ 保持不变;图 3 中用户数量 $m=10\,000$ 保持不变,地点数目 n 从 1 000 ~ 5 000 变化。

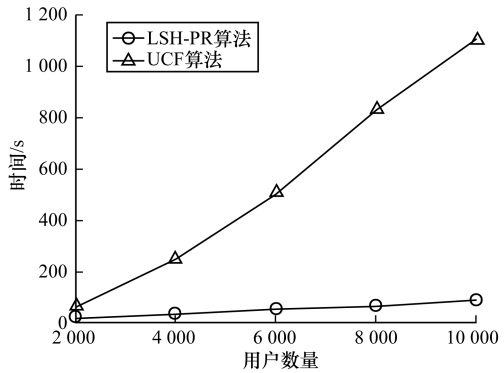

图 2 用户数量变化对运行时间的影响

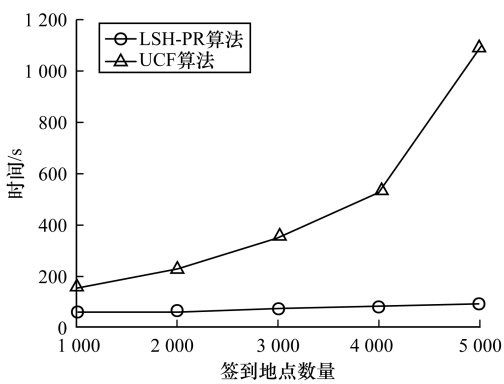

图 3 签到地点数量变化对运行时间的影响

从图 2 可以看出,随着 m 的增大,2 个算法的运行时间都相应变大,但是 LSH-PR 算法一直优于 UCF。其原因是在 LSH-PR 算法中,根据 LSH 的原理,每条用户签到数据都映射到一个哈希桶中,LSH-PR 的时间复杂度为 $O(m)$,而 UCF 算法原理是用户越相似越有可能会访问相同的地点,因此会先计算用户之间的相似度,然后选取相似度高的用户做推荐,由此可知其时间复杂度是 $O(m^2)$。在图 3 中,m 保持不变,随着 n 的逐渐增大,可以看到 LSH-PR 在时间上一直优于 UCF,这时 LSH-PR 的时间复杂度为 $O(mn)$,而 UCF 的时间复杂度为 $O(m^2n)$。因此,本文算法在运行效率上优于传统基于用户的协同过滤推荐算法。

4.2.2 不同推荐方法预测准确度对比

2 种推荐算法的预测准确度对比如图 4、图 5 所示。设置 $T=10$,$r=10$,图 4 中用户数量 m 从 2 000 ~ 10 000 变化,地点数目 $n=5000$ 保持不变;图 5 中用户数量 $m=10\,000$ 保持不变,地点数目 n 从 1 000 ~ 5 000 变化。

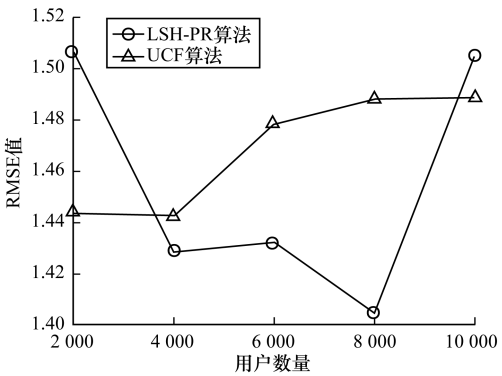

图 4 用户数量变化对 RMSE 值的影响

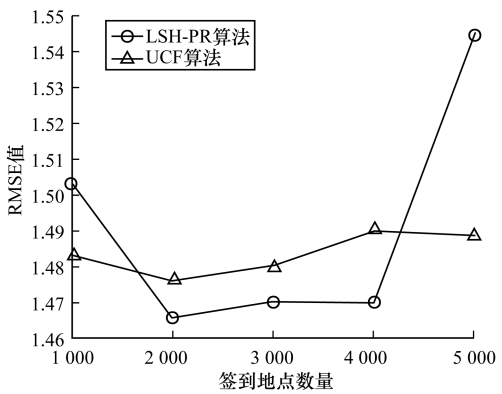

图 5 签到地点数量变化对 RMSE 值的影响

从图 4 可以看出,UCF 算法的预测准确度随着用户数量的增多略微有点下降,但是总体变化不大。LSH-PR 的预测准确度变化幅度比较大,主要是因为 LSH 是一个概率方法,相似度用户集合里的用户只是极大概率的相似用户,并不能保证是准确的相似用户,所以会导致预测准确度的变化。总体来看,

LSH-PR 在多数情况下有较好的预测效果。2 种方法都因为用户数量的增多而导致预测准确度下降的原因是本文使用的真实数据集比较稀疏,用户数量变多反而引入噪声数据,导致预测准确度下降。从图 5 可以看出,UCF 预测结果整体变化幅度不大,LSH-PR 在大多数情况下预测情况优于 UCF,但是因为 LSH-PR 是一个基于概率的算法,并不能确保每次都准确地把相似用户找出来,所以会出现预测情况不如 UCF 的情形。

综合考虑时间消耗和预测准确度,本文提出的算法是一个既可以快速响应用户需求,又可以提供较好预测结果的算法。

4.2.3　不同参数对 LSH-PR 算法的影响

LSH-PR 算法是一种隐私保护的 POI 推荐算法,而 LSH 中哈希表个数和哈希函数个数对用户签到数据映射到哈希散列值有影响,从而影响相似用户集合,本节测试不同 T 和 r 对预测准确度的影响,实验结果如图 6、图 7 所示。设置 m = 10 000 和 n = 5 000保持不变,图 6 中哈希表个数 T 从 10 ~ 20 变化,哈希函数个数 r = 10 保持不变;图 7 中哈希表个数T = 10 保持不变,哈希函数个数 n 从 10 ~ 20 变化。

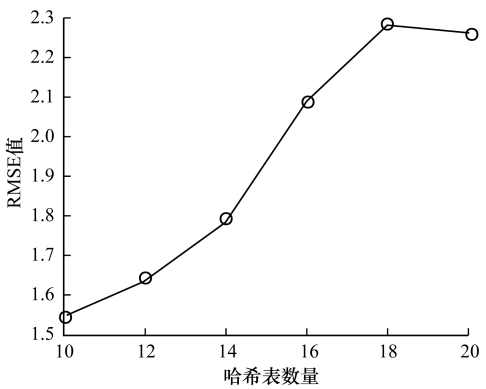

图 6　哈希表数量变化对 RMSE 值的影响

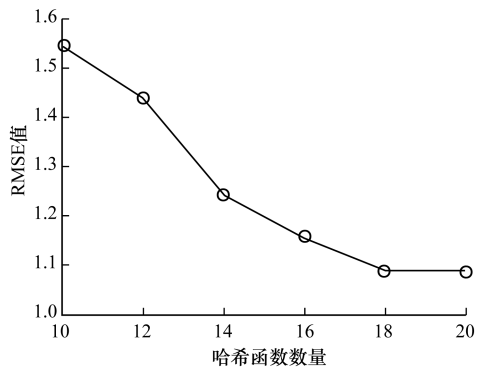

图 7　哈希函数数量变化对 RMSE 值的影响

从图 6 可以看出,随着哈希表个数的增加,LSH-PR 算法的预测准确度下降,这是因为在算法设计中,相似用户集合 *SIM* 是选取不同哈希表中相似用户集合的并集,随着 T 的增大,一些并不相似的用户可能也进入集合 *SIM*,而这些噪声数据影响预测的准确度,因此为提高预测准确度,在实际应用中应该选取较小的哈希表个数。

从图 7 可以看出,随着哈希函数个数的增加,LSH-PR 算法的预测准确度随之提高,这是因为在算法设计中,随着 r 的增大,哈希函数的设计更加严格,只有极为相似的用户才能映射到一个哈希桶中,从而形成相似用户集合 *SIM*,这样就保证了基于这些大概率上的相似用户推荐结果的准确性。因此,为提高预测准确度,在实际应用中应该选取较大的哈希函数个数。

本文算法利用 LSH 和 Paillier 加密技术保护不同社交平台的数据隐私,同时也提供较好的预测结果,实验证明了本文算法的高效性和有效性,在响应时间上优于传统的基于用户的协同过滤推荐算法,在预测准确度上,大部分情况下也显示了该算法优于传统的基于用户的协同过滤算法。

5　结束语

本文提出一种针对分布式平台数据隐私保护的 POI 推荐算法。利用 LSH 和同态加密技术保护各个平台的用户数据隐私,同时满足用户的快速响应需求。实验结果验证了该算法的高效性和有效性。下一步将研究利用差分隐私等其他隐私保护技术来实现分布式平台数据隐私保护的推荐算法。

参考文献

[1] ADOMAVICIUS G, TUZHILIN 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.

[2] BAO J, ZHENG Y, WILKIE D, et al. Recommendations in location-based social networks: a survey [J]. GeoInformatica, 2015, 19(3): 525-565.

[3] YU Y, CHEN X. A survey of point-of-interest recommendation in location-based social networks [C]// Proceedings of the 29th AAAI Conference on Artificial Intelligence. [S.l.]: AAAI Press, 2015: 130.

[4] 冷亚军,陆青,梁昌勇. 协同过滤推荐技术综述[J]. 模式识别与人工智能, 2014, 27(8): 720-734.

[5] BREESE J S, HECKERMAN D, KADIE C. Empirical analysis of predictive algorithms for collaborative filtering [C]//Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. [S. l.]: Morgan Kaufmann Publishers Inc., 1998: 43-52.

[6] GENTRY C. A fully homomorphic encryption scheme[M]. Stanford, USA: Stanford University, 2009.

[7] ERKIN Z, VEUGEN T, TOFT T, et al. Generating private recommendations efficiently using homomorphic encryption and data packing [J]. IEEE Transactions on Information Forensics and Security, 2012, 7(3):

1053-1066.

[8] LIU A, ZHENGY K, LIZ L, et al. Efficient secure similarity computation on encrypted trajectory data[C]// Proceedings of the 31st International Conference on Data Engineering. Washington D. C., USA: IEEE Press, 2015: 66-77.

[9] CANNY J. Collaborative filtering with privacy[C]// Proceedings of 2002 IEEE Symposium on Security and Privacy. Washington D. C., USA: IEEE Press, 2002: 45-57.

[10] LI L, LIU A, LI Q, et al. Privacy-preserving collaborative Web services QoS prediction via YAO's garbled circuits and homomorphic encryption[J]. Journal of Web Engineering, 2016, 15(3): 203-225.

[11] HUANG Y, EVANS D, KATZ J, et al. Faster secure two-party computation using garbled circuits[C]// Proceedings of the 20th USENIX Conference on Security. [S. l.]: USENIX Association, 2011: 35.

[12] NIKOLAENKO V, IOANNIDIS S, WEINSBERG U, et al. Privacy-preserving matrix factorization[C]// Proceedings of 2013 ACM SIGSAC Conference on Computer and Communications Security. New York, USA: ACM Press, 2013: 801-812.

[13] NIKOLAENKO V, WEINSBERG U, IOANNIDIS S, et al. Privacy-preserving ridge regression on hundreds of millions of records[C]//Proceedings of 2013 IEEE Symposium on Security and Privacy. Washington D. C., USA: IEEE Press, 2013: 334-348.

[14] ERKIN Z, VEUGEN T, TOFT T, et al. Generating private recommendations efficiently using homomorphic encryption and data packing[J]. IEEE Transactions on Information Forensics and Security, 2012, 7(3): 1053-1066.

[15] SWEENEY L. k-anonymity: a model for protecting privacy[J]. International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, 2002, 10(5): 557-570.

[16] LI N, LI T, VENKATASUBRAMANIAN S. *t*-closeness: privacy beyond *k*-anonymity and *l*-diversity[C]// Proceedings of 2007 IEEE International Conference on Data Engineering. Washington D. C., USA: IEEE Press, 2007: 106-115.

[17] POLAT H, DU W. Privacy-preserving collaborative filtering using randomized perturbation techniques[C]// Proceedings of the 3rd IEEE International Conference on Data Mining. Washington D. C., USA: IEEE Press, 2003: 625-628.

[18] ZHU J, HE P, ZHENG Z, et al. A privacy-preserving QoS prediction framework for Web service recommendation[C]//Proceedings of 2015 IEEE International Conference on Web Services. Washington D. C., USA: IEEE Press, 2015: 241-248.

[19] ZHANG S, FORD J, MAKEDON F. Deriving private information from randomly perturbed ratings[C]// Proceedings of 2006 SIAM International Conference on Data Mining. [S. l.]: Society for Industrial and Applied Mathematics, 2006: 59-69.

[20] DWORK C. Differential privacy: a survey of results[C]//Proceedings of the 5th International Conference on Theory and Applications of Models of Computation. Berlin, Germany: Springer, 2008: 1-19.

[21] 熊平, 朱天清, 王晓峰. 差分隐私保护及其应用[J]. 计算机学报, 2014, 37(1): 101-122.

[22] MCSHERRY F, MIRONOV I. Differentially private recommender systems: building privacy into the net[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM Press, 2009: 627-636.

[23] JORGENSEN Z, YU T. A privacy-preserving framework for personalized, social recommendations[C]// Proceedings of the 17th International Conference on Extending Database Technology. [S. l.]: EDBT. 2014: 571-582.

[24] GUERRAOUI R, KERMARREC A M, PATRA R, et al. D2P: distance-based differential privacy in recommenders[J]. Proceedings of the VLDB Endowment, 2015, 8(8): 862-873.

[25] LIU X, LIU A, ZHANG X, et al. When differential privacy meets randomized perturbation: a hybrid approach for privacy-preserving recommender system[C]// Proceedings of International Conference on Database Systems for Advanced Applications. Berlin, Germany: Springer, 2017: 576-591.

[26] SHEN Y, JIN H. EpicRec: towards practical differentially private framework for personalized recommendation[C]//Proceedings of 2016 ACM SIGSAC Conference on Computer and Communications Security. New York, USA: ACM Press, 2016: 180-191.

[27] GIONIS A, INDYK P, MOTWANI R. Similarity search in high dimensions via hashing[C]//Proceedings of the 25th International Conference on Very Large Data Bases. [S. l.]: Morgan Kaufmann Publishers Inc., 1999: 518-529.

[28] 史世泽. 局部敏感哈希算法的研究[D]. 西安: 西安电子科技大学, 2013.

[29] 陈志伟, 杜敏, 杨亚涛, 等. 基于 RSA 和 Paillier 的同态云计算方案[J]. 计算机工程, 2013, 39(7): 35-39.

编辑 司森森