基于协同过滤的位置感知推荐

李 贵 陈盛红 韩子阳 李征宇 孙 平 孙焕良
(沈阳建筑大学信息与控制工程学院 沈阳 110000)

摘 要 不同地区的用户兴趣不同,并且当推荐物品具有位置属性时,用户更加倾向于离自身较近的物品。根据用户和物品的位置信息来捕获用户兴趣能有效地提高个性化推荐精度。为了有效处理用户和物品的位置信息,在推荐系统中引入金字塔模型(PS)来实现用户分区和用户旅行代价的计算,提出了基于金字塔模型的协同过滤算法(PMCF),来生成对用户的Top-N物品推荐。使用MovieLens数据集、Foursquare数据集和Synthetic数据集来分别评估算法的有效性,实验表明,所提出的算法的准确度要高于传统的推荐算法。

关键词 位置感知,金字塔模型,协同过滤,推荐系统

中图法分类号 TP301.6 **文献标识码** A

Location-aware Recommendation Based on Collaborative Filtering

LI Gui CHEN Sheng-hong HAN Zi-yang LI Zheng-yu SUN Ping SUN Huan-liang
(Faculty of Information & Control Engineering, Shenyang Jianzhu University, Shenyang 110000, China)

Abstract Users have different interests in different regions, and when recommended items are spatial, users tend to travel a limited distance when visiting these venues. Accurately capturing user preferences according to the users' and items' location can improve the precision in recommender systems. To effectively deal with users' and items' location information, this paper introduced Pyramid Model(PM) in recommender systems for realizing users' partitioning and calculating travel penalty, and presented a collaborative filtering recommendation algorithm based on Pyramid model(PMCF) to generate Top-N recommend. MovieLens, Foursquare and Synthetic data set were quoted to evaluate the effectiveness of the algorithm. Experimental results show our algorithm has significant improvements in terms of effectiveness measured through precision.

Keywords Location-aware, Pyramid model, Collaborative filtering, Recommender systems

1 引言

推荐系统能够在信息过载的环境中帮助用户发现令其感兴趣的物品信息,并将信息推荐给他们,例如京东网上商城等电子商务网站的产品推荐,Netflix等视频网站的视频推荐,谷歌和新浪等媒体网站的新闻推荐。随着移动设备的发展,用户的位置信息很容易获取,用户可以通过移动设备对其感兴趣的物品产生评分——位置感知评分。例如,Foursquare和Facebook等基于位置的社交网络通过用户在目的地(如餐厅)的"签到"服务,得到用户的位置感知评分,然后通过用户的评分信息产生基于位置的推荐。

不同地区的用户兴趣不同,即用户偏好具有区域性。例如,在2月份,位于哈尔滨的用户对冬装更感兴趣,而位于三亚的用户可能对春装更感兴趣。因此,根据用户的位置信息来捕获用户的兴趣偏好在现实世界应用中是非常重要的。但是传统的位置感知推荐系统只是简单地根据每个用户的位置来计算用户之间的相似度,这样的计算复杂度高,不能有效地体现用户偏好的区域性。

当推荐的物品具有位置属性时,用户更加倾向于离自身距离较近的物品,即用户旅行的区域性。根据Foursquare数据集的分析,45%的用户选择距离10英里或更近的地点,75%的用户选择距离50英里或更近的地点。这表明推荐系统在推荐空间物品时,应优先推荐距离用户较近的物品。当距离用户较远且用户必须去时,推荐系统就没有意义了。

为了有效处理用户和物品的位置信息,本文在推荐系统中引入金字塔模型(PS)来实现用户分区和计算用户旅行代价,提出了基于金字塔模型的协同过滤算法(PMCF),来生成对某个用户的Top-N物品推荐。

网络中具有位置信息的用户评分一般分为3种类型[1]:

1)非空间物品的位置评分。通过四元组(用户,用户位置(ulocation),评分,物品)来表示。评分信息中具有用户位置信息,但不具有物品位置信息,例如一个用户在家里用手机对网络中某一部电影进行评分;

2)空间物品的非位置评分,通过四元组(用户,评分,物品,物品位置(ilocation))来表示。评分信息中具有物品位置信息,但不具有用户位置信息,例如一个用户在未知位置对某

本文受国家自然科学基金(61070024),辽宁省自然科学基金(2014020068)资助。

李 贵(1964-),男,博士,教授,主要研究方向为Web数据挖掘与信息集成、分布对象技术、软件工程,E-mail:Ligui21c@sina.com;陈盛红(1991-),男,硕士生,主要研究方向为Web数据挖掘和推荐系统。

一酒店进行评分；

3)空间物品的位置评分，通过五元组(用户，用户位置(ulocation)，评分，物品，物品位置(ilocation))来表示。评分信息中用户和物品都具有位置信息，例如一用户在家里对某一餐厅进行评分。

传统的评分通过三元组(用户，评分，物品)来表示，因为评分信息中没有位置信息，所以不包含在这些类别中。

2 相关工作

本节主要介绍基于位置推荐的相关研究工作。

2.1 传统推荐

大多数推荐系统使用的评分数据是传统评分数据，用三元组(用户，评分，物品)来表示。文献[2]将上下文属性(如天气，交通)融入推荐系统中来考虑位置信息，但只是简单地利用用户和物品间的位置距离来给用户推荐感兴趣的物品。文献[3]介绍了 Netflix 的“本地偏好”推荐列表，列表中包含同一城市的用户特别喜好的物品。这个“本地偏好”推荐列表是根据一个城市的电影总体租赁数据来创建的，但是这个列表对于每个用户都是相同的，并不具有独特性。本文所提出的 PMCF 算法是利用用户的偏好区域性和旅行区域性来给用户产生个性化位置感知推荐。

2.2 位置感知推荐

目前位置感知推荐主要采用两种技术：

(1)KNN 技术[4]或聚类 KNN 技术，通过简单检索推荐用户位置附近的 K 个对象，但没有考虑用户个性化需求。

(2)偏好建模，skyline 模型[5]和基于位置的 Top-N 推荐模型[6]，需要用户提供明确的偏好条件。

文献[7]提出了基于位置的排名，给定一个用户的位置信息和偏好类型(如餐厅)，系统通过分析用户的历史记录来产生一个 Top-N 物品推荐列表。单纯基于位置的排名并不能实现个性化推荐，两个用户在同一位置通过基于位置的排名系统得到的推荐结果是相同的。文献[8]介绍了 CityVoyager 系统，该系统通过挖掘用户的 GPS 轨迹数据来发现用户常去的购物场所，预测用户未来可能光顾的地点并产生推荐结果。文献[9]通过挖掘包含用户标签的 GPS 轨迹数据，检测在同一城市的用户喜好的活动场所，如艺术展览或学校附近的餐厅。系统通过这种数据来解决两个问题：(1)给定一个活动类型，返回这种活动在这个城市发生的位置；(2)给定一个明确空间区域，返回该区域内的所有活动。这两个问题都是通过挖掘用户的 GPS 轨迹数据产生的推荐结果来解决的。文献[10]提出了基于用户地理位置测量的协同过滤算法，通过用户同城社交网络好友的评分来产生推荐结果。本文所提出的 PMCF 算法通过分析用户的偏好区域性和旅行区域性，在系统中引入了金字塔模型(PS)来实现用户分区和计算用户旅行代价，提出了基于金字塔模型的协同过滤算法来挖掘用户隐形偏好，产生个性化推荐结果。

3 金字塔模型

3.1 数据类型

金字塔模型建模所采用的数据必须包含位置信息，如用户位置信息或物品位置信息。本节通过两种数据类型来分别创建金字塔模型，这两种数据类型分别为非空间物品的位置评分数据和空间物品的非位置评分数据。非空间物品的位置评分数据由四元组(用户，用户位置，评分，物品)表示，可以通过金子塔模型实现用户分区。空间物品的非位置评分数据由四元组(用户，评分，物品，物品位置)表示，可以通过金子塔模型实现区域旅行代价的计算。

3.2 金字塔模型结构

PMCF 算法采用的金字塔模型如图 1 所示。模型分成 H 级(即金字塔模型的高度)。每个等级中的区域网格所代表的现实地理区域规模不同，网格所在的等级越低，网格所代表的现实地理区域规模越大。第 h 级上的网格单元是第 $h-1$ 级的网格单元的子类，即第 h 级上存在 m 个网格单元在第 $h-1$ 级上有共同的父类网格单元，其中 m 远小于第 h 级网格单元总数。如图 2 所示，用金字塔模型来代表中国用户/物品的地理空间区域，那么金字塔根部(0 级)仅有的一个网格单元代表整个中国地理区域空间。金字塔模型中 1 级内的每个网格单元都是 0 级单元的子类，且代表一个省(或直辖市)地理区域空间。金字塔模型中 1 级内网格单元子类都位于 2 级内，并且每个 2 级网格单元是 1 级单元的子类，且代表着一个地级市。如果需要对用户位置进行更加精细的划分，依此金字塔模型可设置 3 级或更高级来表示更精细的地理位置空间。值得注意的是，金字塔模型的根区域(0 级)代表了“传统”的基于物品的协同过滤模型(即对整个空间内的所有用户/物品进行分析)。

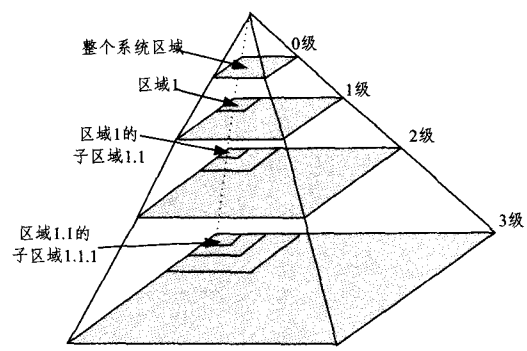

图 1 金字塔数据结构

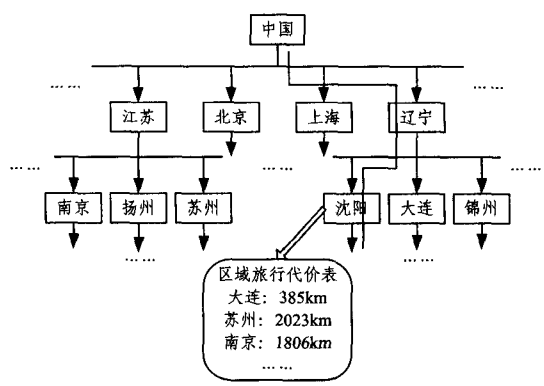

图 2 一个金字塔模型简单实例

PMCF 算法通过非空间物品的位置评分数据来创建用户金字塔模型，它是根据四元组(用户，用户位置，评分，物品)中用户位置来对用户进行区域划分，对于同一区域的用户信息用三元组(用户，评分，物品)表示，然后利用基于物品的协同过滤技术来处理单个区域内的用户评分，形成区域推荐结果。

PMCF 算法通过空间物品的非位置评分数据来创建物品金子塔模型，根据四元组(用户，评分，物品，物品位置)中的物品位置对物品进行区域划分，然后在每个区域单元中创建一个区域旅行代价表，来存储这个区域到同等级其它区域的旅行代价。如图 2 所示，沈阳这个区域内的旅行代价表存储了沈阳到等级 2 上所有城市的旅行代价。区域旅行代价的计算将在 4.3 节做详细介绍。

本文采用金字塔模型是因为它是“空间分割”结构，可以确保完全覆盖给定的空间。为达到本文目的，“数据分割”结构(如 R 树)不太理想，它的索引数据点不能确保完全覆盖所有给定空间。

4 基于金字塔模型的协同过滤算法(PMCF)

4.1 协同过滤算法

PMCF 算法采用基于物品的协同过滤(简称 ItemCF)技术作为主要的推荐技术，因为 ItemCF 技术是目前业界应用最多的技术。无论是亚马逊，还是 Netfix、Hulu、YouTube，其推荐算法的基础都是该技术。

ItemCF 技术主要分为两步：

1)计算物品之间的相似度；

2)根据物品的相似度和用户的历史行为为用户生成推荐列表。

4.1.1 物品间相似度的计算

这节计算物品 i 和物品 j 的相似度 $\mathrm{sim}(i,j)$，这里 i 和 j 至少被同一用户做出过评分。本文采用的 ItemCF 技术使用修正的余弦相似度[11](adjust cosine similarity)来计算物品间的相似度。两个物品如果被同一用户打过相同的评分，那么其物品间的相似度比被同一用户打过不同评分的两个物品之间的相似度高，本文通过使用同一用户对两个物品的评分差值的倒数来修正余弦相似度，计算公式为：

$$\mathrm{sim}(i,j)=\frac{\sum_{u\in U}(r_{u,i}-\overline{r_u})(r_{u,j}-\overline{r_u})}{\sqrt{\sum_{u\in U}(r_{u,i}-\overline{r_u})^2\sum_{u\in U}(r_{u,j}-\overline{r_u})^2}}*\frac{1}{1+\log(1+\sum_{u\in U}|r_{u,i}-r_{u,j}|)} \tag{1}$$

式中，$N(i)$是给物品 i 打分的用户集合，$U=N(i)\cap N(j)$是对物品 i 和物品 j 都评分的用户集合。

4.1.2 生成推荐列表

给定一个用户 u，PMCF 算法通过计算用户 u 对物品 i 的预测评分 $P_{(u,i)}$来判断用户 u 对物品 i 的喜好程度，这里物品 i 都是没被用户 u 评分的。

$$P_{(u,i)}=\overline{r_i}+\frac{\sum_{j\in S(i,k)\cap N(u)}\mathrm{sim}(i,j)*(r_{u,j}-\overline{r_i})}{\sum_{j\in S(i,k)\cap N(u)}|\mathrm{sim}(i,j)|} \tag{2}$$

式中，$S(i,k)$是与物品 i 最相似的 K 个物品集合。$N(u)$是被用户 u 打分的物品集合。$\mathrm{sim}(i,j)$是物品 i 与物品 j 的相似度。PMCF 算法会给用户 u 推荐 $P_{(u,i)}$值最高的 N 个物品。

4.2 非空间物品的位置评分

考虑用户偏好具有区域性，本文通过金字塔模型将用户 u 的位置 L 进行层次划分，在 h 等级的区域内生成推荐列表 $N(u,l_h)$，然后将用户在所有不同等级区域中产生的推荐列表按一定的权重进行线性相加得到用户 u 最终的推荐列表 $N(u)$，即用户位置感知推荐列表。

如图 3 所示，给定一个用户 u 和位置 L(如中国辽宁省沈阳市浑南新区浑南东路)，将用户位置 L 在金字塔模型进行分区，图 3 被分为中国、辽宁、沈阳、浑南新区 4 个等级区域空间。用户位置所在的每级区域空间根据 ItemCF 技术生成区域推荐列表 $N(u,l_h)$，其中 u 为查询用户，h 为金字塔模型中的级数。然后将用户 u 所有的区域推荐列表 $N(u,l_h)$按一定权重 ω_h 进行线性相加得到最终推荐列表 $N(u)$，提取偏好值最高的 N 个物品推荐给用户 u，形成 Top-N 推荐。本文以预测评分做实验来评测算法的有效性，最终得到用户 u 对物品 i 的偏好值为用户 u 对物品 i 的预测评分 $P_{(u,i)}$。物品 i 与物品 j 在 h 级空间区域内的区域相似度 $\mathrm{sim}(i,j,l_h)$的计算如式(3)所示。

$$\mathrm{sim}(i,j,l_h)=\frac{\sum_{u\in U\cap l_h}(r_{u,i}-\overline{r_u})(r_{u,j}-\overline{r_u})}{\sqrt{\sum_{u\in U\cap l_h}(r_{u,i}-\overline{r_u})^2\sum_{u\in U}(r_{u,j}-\overline{r_u})^2}}*\frac{1}{1+\log(1+\sum_{u\in U}|r_{u,i}-r_{u,j}|)} \tag{3}$$

式中，$u\in U\cap l_h$ 表示用户 u 所在 h 级的区域空间内对物品 i 与物品 j 都评过分的用户集合。通过区域相似度 $\mathrm{sim}(i,j,l_h)$可以计算出用户 u 对物品 i 在 h 级区域空间内的预测评分 $P_{(u,i,l_h)}$，如式(4)所示。

$$P_{(u,i,l_h)}=\overline{r_{i,l_h}}+\frac{\sum_{j\in S(i,k)\cap N(u)}\mathrm{sim}(i,j)*(r_{u,j}-\overline{r_{i,l_h}})}{\sum_{j\in S(i,k)\cap N(u)}|\mathrm{sim}(i,j)|} \tag{4}$$

式中，$\overline{r_{i,l_h}}$为物品 i 在 h 级区域空间内的用户给的平均评分。将区域预测评分 $P_{(u,i,l_h)}$进行线性加权融合得到用户 u 对物品 i 的最终评分 $P_{(u,i)}$，如式(5)所示。最后将 $P_{(u,i)}$值最大的 N 个物品加入推荐列表中，产生用户 u 的 Top-N 推荐。

$$P(u,i)=\sum P(u,i,l_h)*w_h \tag{5}$$

其中，

$$\begin{cases}w_h\in[0,1]\\ w_1+w_2+\cdots+w_h=1\end{cases} \tag{6}$$

式中，w_h 为 h 级区域空间的区域权重值，控制每个等级区域空间产生的区域评分对最终评分的影响。本文通过实验测试得到 PMCF 算法产生最优推荐时区域权重 w_h 的比例。

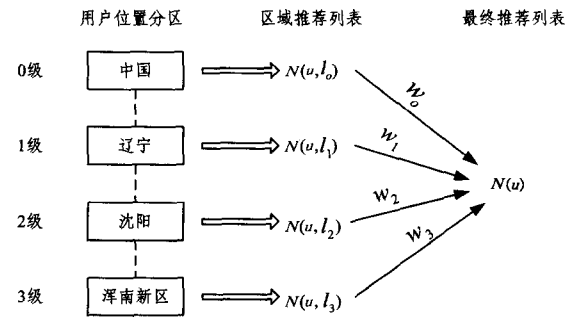

图 3 基于金字塔模型的协同过滤算法

4.3 空间物品的非位置评分

这部分介绍 PMCF 算法利用空间物品的非位置评分产生推荐，评分由四元组(用户，评分，物品，物品位置)表示。当推荐的物品具有位置属性时，用户更加倾向于离自身距离较近的物品，即用户旅行具有区域性。系统利用旅行代价来限制用户选择空间物品所在的区域，在合理的旅行距离上产生位置感知推荐。旅行代价是通过惩罚距离用户较远位置的物

品来影响位置感知推荐结果。空间物品是相对静止的，如餐厅，不会随便改变位置，因此本文不考虑系统中现有物品因位置改变而改变旅行代价。

推荐系统首先忽略物品的位置信息，利用 ItemCF 技术来生成用户 u 对物品 i 的预测评分 $P(u,i)$，但最终物品 i 在用户 u 的推荐列表中的推荐值定义为：

$$RecScore(u,i)=P(u,i)-TravelPenalty(u,i) \quad (7)$$

式中，$TravelPenalty(u,i)$ 表示用户 u 到物品 i 的旅行代价。计算 $TravelPenalty(u,i)$ 的基本思想是对物品 i 与用户 u 之前评分的所有物品的位置计算距离的平均值，然后将所求的平均值归一化到评分取值的相同范围内，如[0,5]。旅行代价的计算有两种方法：

1)KNN[4]：一种精确的在线方法。给定物品 i 的位置 L，KNN 算法能返回距离物品 i 旅行代价最小的物品集合。KNN 技术的优点是提供了用户到每个候选推荐物品之间的准确旅行距离，缺点是计算复杂度高。例如，在欧氏空间上使用 KNN 技术，运行时检索单个物品的时间复杂度[4]为 $O(k+\log N)$，这里 N 和 k 分别是物品总数和物品检索数量。

2)区域旅行代价：一种离线启发式方法，计算效率较高，但结果不太精确。通过金字塔模型根据物品位置来实现物品区域划分，通过线下预先计算同一等级中两个区域单元间的旅行代价来表示两个区域单元内的用户间旅行代价。每个区域单元 c 中存在一个旅行代价表，存储了区域单元 c 到同一等级的其它区域单元的旅行代价。区域单元 c 到区域单元 d 的旅行代价为区域单元 c 中的所有物品位置到区域单元 d 中所有物品位置的平均旅行距离。

本文提出的 PMCF 算法结合两种方法来计算 $TravelPenalty(u,i)$。首先选择区域计算所在的金字塔等级 h（如 $h=2$），如果两个物品的位置在等级 h 上的同一区域单元中，就通过 KNN 技术计算其旅行代价，否则就将两个物品所在的区域单元的区域旅行代价作为它们之间的旅行代价。但是为了避免计算用户 u 对每个未评分的物品 i 的 $P_{(u,i)}$，本文首先根据旅行代价的大小对用户 u 未评分的所有物品进行递增排序，然后由小到大依次进行计算，最后在查询处理中使用提前终止法则[12]，如算法 1 所示。

算法 1 为实现空间物品推荐的伪代码。输入项为查询用户 u 和限制条件 N。返回 Top-N 推荐列表 R。算法首先通过 k 近邻(KNN)技术和区域旅行代价技术计算出旅行代价最小的 N 个物品并加入推荐列表 R 中。然后设置推荐列表 R 的最低推荐值 LowestRecScore(算法 3－8 行)来实现算法的初始化。对于每个物品 i，将其最大评分 MAX_RATING 减去其旅行代价得到最大预测评分 MaxPossibleScore(算法 12 行)。如果 MaxPossibleScore 不大于最低推荐值 LowestRecScore，算法立即终止计算其预测评分和推荐分值(算法 13－15)。如果 MaxPossibleScore 大于最低推荐值 LowestRecScore，算法通过式(7)计算其推荐分值 RecScore，并更新推荐列表 R 的最低推荐值 LowestRecScore(算法 16－20)。

算法 1 空间物品推荐的伪代码

输入：用户 u 和限制条件 N

输出：用户 u 的 Top-N 推荐

```
1. Function LARS _SpatialItems(User u,Limit N)
2. /* 填充包含 N 个物品的推荐列表 R */
3.   R←Ø/* 将推荐列表 R 设置为空集 */
4. for (N iterations)  do
5.   i←Retrieve the item with the next lowest travel penalty
6.   Insert i into R ordered by RecScore(u,i) computed by Equation 7
7. end for
8. LowertRecScore ← RecScore of the Nth object in R
9. /* 依此检索每个物品的旅行代价值 */
10. while there are more items to process do
11.   i←Retrieve the next item in order of penalty score
12.   MaxPossibleScore←MAX_RATING－i. penalty
13.   if MaxPossibleScore ≤ LowertRecScore then
14.     return R /* 提前终止，结束查询 */
15.   end if
16.   RecScore(u,i)←P(u,i)－i. penalty
17.   if RecScore(u,i) >LowertRecScore then
18.     Insert i into R ordered by RecScore(u,i)
19.       /* 将物品 i 加入列表 R 中 */
20.     LowertRecScore← RecScore of the Nth object in R/* 更新最低推荐值 */
21.   end if
22. end while
23. return R
```

4.4 空间物品的位置评分

这节介绍 PMCF 算法利用空间物品的位置评分产生推荐，评分由五元组(用户，用户位置，评分，物品，物品位置)表示。当用户和物品都具有位置属性时，可以通过用户分区和旅行惩罚来分别分析用户的偏好区域性和旅行区域性。物品 i 在用户 u 的推荐列表中的推荐值定义为：

$$RecScore(u,i)=P(u,i,l_h)-TravelPenalty(u,i) \quad (8)$$

式中，$P(u,i,l_h)$ 是在用户 u 的位置所在第 h 级区域空间内用户 u 对物品 i 的预测评分，计算过程如 4.2 节所述。h 为用户设定的区域等级，如用户选择沈阳市($h=2$)作为其物品位置选择范围，$TravelPenalty(u,i)$ 是用户 u 到物品 i 的旅行代价，计算过程如 4.3 节所述。查询处理也使用算法 1 产生推荐，但唯一不同的是，需将算法 16 行中的 $P(u,i)$ 改为区域评分 $P(u,i,l_h)$。

5 金字塔模型的维护

5.1 推荐查询处理

在基于金字塔模型的推荐系统中，给定一个用户或物品的位置 L 和金字塔模型的级数限制条件 H(一般设置 $H=4$，即为 4 等级金字塔模型)，本文推荐系统需要执行两个查询处理步骤：

(1)将用户或物品的位置 L 在金字塔模型中进行分层区域划分，找到包含用户或物品的位置 L 的最低等级的区域单元。

(2)在包含用户位置的每级区域单元中，使用 ItemCF 技术生成区域推荐结果。在包含物品位置的每级区域空间中使用区域旅行代价技术计算同一等级中所有区域单元间的旅行代价，使用 KNN 技术计算同一区域单元中每个物品间的旅行代价。

5.2 推荐更新

当用户查询请求发出时，系统就开始计算并产生推荐结果给用户。PMCF 通过用户位置更新来监控其运动轨迹，只

要用户的运动轨迹不跨出其在金字塔模型中的最小区域单元空间，就不需要更新用户的推荐结果。当用户的运动轨迹跨出其在金子模型中的最小区域单元空间时，系统就会更新这个区域推荐结果，进而影响最终推荐。金字塔模型中更高级别的区域单元需要维持较大的空间区域，用户运动轨迹跨越的其空间界限的机会较少，其区域推荐结果更新次数也就较少。

5.3 数据结构维护

PMCF 算法使用位置感知评分来构建 H 级金字塔模型，模型的等级 H 最初是根据用户位置或物品位置的最小区域信息所设定的(如街道，村)。在金字塔模型构建期间，如果该区域的用户或物品量过少，会导致该区域内的数据过于稀疏，进而难以建立协同过滤模型或者区域旅行代价计算误差较大，那么就需要将这个区域和其全部子类区域都合并到其父类区域中。由于发达城市的用户过于集中，可能导致金字塔中最低等级区域空间包含的用户数过大，因此可以将该区域进行更精细的划分，产生子类空间区域单元。

随着时间的推移，新的用户、新的评分和新的物品都会被添加到系统中。这些新数据会更新金字塔模型的数据规模，加大单元中协同过滤模型的规模，从而改变区域单元中的推荐结果。考虑到这些变化，本文推荐系统需要对每个基础区域单元进行维护。维护就是当一个基础区域单元 c 内增加了 $M\%$的新评分就进行推荐更新，$M\%$是区域单元 c 接收的新评分与现存的评分数量比，本文将 M 设定为 10。由于协同过滤是一种成熟的模型，并在很多数据模型中得到应用，且需要很多的数据的更新才能改变 Top-N 推荐，因此系统不需要时时维护。金字塔模型维护包含两个重要步骤：模型重建和合并/拆分维护。

1)模型重建。即重建区域单元 c 中基于物品的协同过滤模型和重新计算区域单元 c 的区域旅行代价表。重建就是将新的评分信息添加到模型中，模型重建也叫模型更新。

2)合并/拆分维护。在重建单元 c 后，系统调用合并/拆分维护步骤，根据用户和物品数量的平衡情况，决定合并/拆分单元。一方面，如果算法检测到等级 h 上的区域单元 c 在等级 $h+1$ 上存在一个子类区域单元 q 中用户数过于稀疏(如用户数小于 100)，且 q 不存在子类区域单元，那么系统将区域单元 q 合并到父类单元 c 中。另一方面，如果等级 h 上区域单元 c 在等级 $h+1$ 没有子类区域单元，且单元 c 中用户数据量超大(如用户数大于 10000)，那么系统根据位置划分将区域单元 c 拆分为位于等级 $h+1$ 上的多个子类区域单元。

金字塔模型的更新具有下列特征：

1)更新可以完全在线下完成，即当部分金字塔中单元正在更新时，系统继续使用"旧"金字塔模型来产生推荐；

2)更新并不需要重建整个金字塔模型，只需要重建一个区域单元；

3)只有当 $M\%$的新评分加入金字塔单元后才执行更新，即更新会分批次进行操作。

6 实验结果

6.1 数据集描述

MovieLens 数据集：一个包含非空间物品的位置评分真实数据集。这个数据集中包含了 814 个用户对 1668 部电影的 87025 个评分。每个电影的评分和对其评分的用户邮政编码相关联，使得评分信息中具有用户位置信息。为了便于实验，本文从用户评分数据库中选择 6000 条评分数据作为实验数据集，实验数据集中共包含 145 个用户和 805 部电影，其中每个用户至少对 20 部电影进行了评分。

Foursquare 数据集：一个包含空间物品的非位置评分的真实数据集。本文采用的 Foursquare 数据集包含 18107 个用户、跨越美国的 43063 个地点和 2073740 个签到。

Synthetic 数据集：本文实验中使用的 Synthetic 数据集包含 2000 个用户、1000 个物品和 500000 个评分。实验在明尼苏达州随机生成用户和物品位置，是一个包含空间物品的位置评分的真实数据集。数据集中用户对物品的评分是 0 到 5 之间。

实验中将所有数据集分成两份：一份是训练数据集，占总数据集的 80%，用来建立推荐模型；另一份是测试数据集，占总数据集的 20%，用来测试评分预测准确性。

6.2 度量标准

评价推荐系统推荐质量的度量标准主要包括统计精度度量方法和决策支持精度度量方法两类[11]。统计精度度量方法中的均方根误差 RMSE(Root Mean Square Error)可以直观地对推荐精度进行度量，是一种常用的推荐精度度量方法，本文采用均方根误差 RMSE 作为度量标准。均方根误差 RMSE 通过计算预测的用户评分与实际的用户评分之间偏差的均方根来度量评分预测的准确性，RMSE 越小，推荐精度越高。设预测的用户评分集合为$\{p_1, p_2, \cdots, p_N\}$，对应的实际用户评分集合为$\{q_1, q_2, \cdots, q_N\}$，则均方根误差 RMSE 定义[1]为：

$$RMSE=\frac{\sqrt{\sum_{i=1}^{N}(p_i-q_i)^2}}{N} \tag{9}$$

6.3 实验结果

为了验证本文提出基于金字塔模型的协同过滤算法(PMCF)在 3 种评分数据集上的有效性，分别与基于物品评分预测的推荐算法(IR-Based CF)和基于奇异值分解的推荐算法(SVD-Based CF)在 MovieLens 数据集、Foursquare 数据集和 Synthetic 数据集中分别进行比较。通过比较 3 种推荐算法在评分预测的 RMSE 大小，分析算法的优越性。实验中设定 $N=10$。

6.3.1 在 MovieLens 数据集上的实验结果

这部分讨论 PMCF 算法在非空间物品的位置评分数据集(MovieLens 数据集)上的有效性。

6.3.1.1 参数 w_n 的影响

本节分析参数 w_n(式(5)和式(6))对 PMCF 算法的影响。在 PMCF 算法中，w_n 控制每个等级的区域预测评分对最终评分的影响。如果 $w_1=0$，表示最终评分不考虑等级 1 上区域评分影响。$w_1=1$，表示最终评分只受等级 1 上区域评分影响，即 $P(u,i)=P(u,i,l_1)$。实验中采用 3 层金字塔模型，w_n 取值如表 1 所列，实验结果如图 4 所示。

表 1 w_n 取值和比例

w_1	0	0	1	1/3	1/6	1/2
w_2	0	1	0	1/3	1/3	1/3
w_3	1	0	0	1/3	1/2	1/6
$w_0:w_1:w_2$	0:0:1	0:1:0	1:0:0	1:1:1	1:2:3	3:2:1

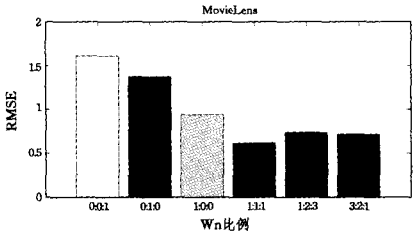

图 4　在 MovieLens 数据集上，w_n 对 PMCF 算法的影响

由图 4 可知，w_n 比例为 1:1:1，即 $w_0=w_1=w_2=1/3$ 时，RMSE 最小，是 FSCF 算法的评分预测精度也最高。当 $w_0:w_1:w_2=0:0:1$、$0:1:0$ 或 $1:0:0$ 时，RMSE 都较大，表明如果只考虑某个单一区域的区域预测评分，评分预测精度较低。当 $w_0:w_1:w_2=1:2:3$ 或 $3:2:1$ 时，RMSE 也无法达到最小值，表明如果不平衡不同等级区域的影响，评分预测精度也无法达到最高。综合实验考虑，当每个等级区域评分所占比例相同，即 $w_0=w_1=\cdots=w_n=1/n$ 时，评分预测精度最高，PMCF 算法评分预测结果最优。接下来实验中就设置 $w_0=w_1=\cdots=w_n=1/n$。

6.3.1.2　金字塔模型等级 H 的影响

不同等级的金字塔模型产生的推荐结果是不同的，实验中通过比较不同 H 值的情况下，PMCF 算法和其它推荐算法的 RMSE 值来评测 PMCF 算法有效性。金字塔模型等级 H 从 0 增加到 4，实验结果如图 5 所示。

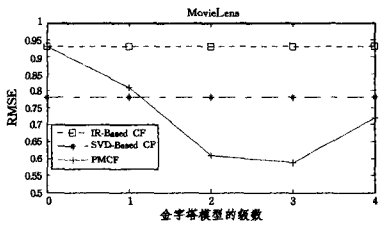

图 5　在 MovieLens 数据集上，算法推荐精度比较

从图 5 可以看出，IR-Based CF 和 SVD-Based CF 的 RMSE 值是个常数，因为这两种算法不是基于金字塔模型的，所以其 RMSE 值不随金字塔模型级数 H 变化而变化。当金字塔模型的级数为 3 时，PMCF 评分预测的 RMSE 值最小，且小于其它推荐算法的 RMSE。也就是当金字塔为 3 级结构时，PMCF 的推荐精度最高，优于其它推荐算法。这也表明，通过金字塔模型进行用户分区有利于提高推荐系统的精度，但如果金字塔模型的级数过大，会导致底层区域单元用户过于稀疏，产生“饥饿推荐”，进而影响推荐精度。

6.3.2　在 Foursquare 数据集上的实验结果

这部分讨论 PMCF 算法在空间物品的非位置评分数据集(Foursquare 数据集)上的有效性。在不同等级区域中将 KNN 技术和区域旅行代价技术相结合，产生的推荐结果精度和查询响应时间都不同，实验结果如图 6 和图 7 所示。

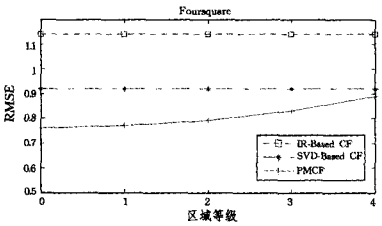

图 6　在 Foursquare 数据集上，算法推荐精度比较

从图 6 可以看出，在 Foursquare 数据集上，PMCF 算法在不同等级区域上的 RMSE 值都小于 IR-Based CF 和 SVD-Based CF 的 RMSE 值，这表明在给用户推荐具有位置信息的物品时，考虑用户旅行代价能提高推荐系统的精度。从图中也可以看出，PMCF 算法的 RMSE 值随着金字塔区域等级 H 的增加而增加，因为随着区域等级 H 的增加，同一等级中包含的区域单元越多，区域旅行代价在旅行代价计算中所占比重越大，旅行代价的计算结果就越不精确，导致 RMSE 的值越大。IR-Based CF 和 SVD-Based CF 的 RMSE 值是个常数，这是因为这两种算法不是基于金字塔模型的，所以其 RMSE 值不随金字塔模型级数变化而变化。

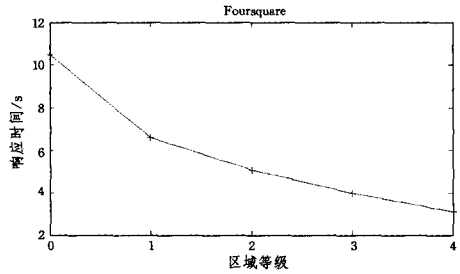

图 7　不同等级下的查询响应时间

从图 7 可以看出，随着区域等级 H 的增加，推荐系统的查询响应时间会降低。因为区域等级越高，在旅行代价计算中区域旅行代价计算所占比重越大，KNN 计算所占比重越小，在线计算时间就越短。而区域旅行代价是线下预先计算的，消耗的时间可以忽略不计，需要考虑的主要是 KNN 计算消耗时间。

综合考虑推荐精度和响应时间，在旅行代价计算中，本文选择在金子塔模型的 2 级区域中通过 KNN 技术和区域旅行代价技术来计算旅行代价。

6.3.3　在 Synthetic 数据集上的实验结果

这部分讨论 PMCF 算法在空间物品的位置评分数据集(Synthetic 数据集)上的有效性。h 为用户自行选择的区域等级，从空间物品的非位置评分在 Foursquare 数据集上的实验结果和非空间物品的位置评分在 MovieLens 数据集上的实验结果可以看出，当 l_h 设定为 l_2 时，产生的推荐结果精度较高且查询响应时间较短。在下面实验中将 l_n 设定为 l_2。实验结果如图 8 所示。

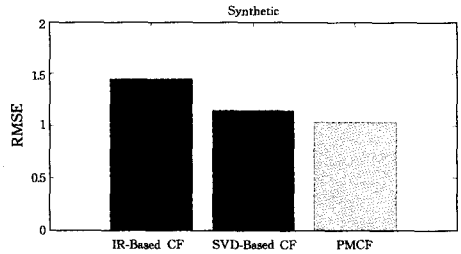

图 8　在 Synthetic 数据集上，算法推荐精度比较

从图 8 可以看出，在 Synthetic 数据集上，PMCF 的 RMSE 值小于 IR-Based CF 和 SVD-Based CF 的 RMSE 值，这表明 PMCF 算法相对较优。这也表明当用户和物品都具有位置信息时，通过金子塔模型分别对用户和物品进行分区能有效提高推荐系统的精度。

结束语　用户偏好受用户自身位置和物品位置的影响，捕获和利用两者位置信息对于位置感知推荐具有很大的实际意义。本文提出了基于金字塔模型的协同过滤算法(PMCF)，通过 3 种基于位置的评分数据(非空间物品的位置评分、空间物品的非位置评分、空间物品的位置评分)来产生位置感知推荐。PMCF 算法通过金字塔模型(PM)来实现用户

分区和计算用户旅行代价。在真实与合成的数据集上的实验表明，PMCF 算法可以有效利用用户和物品的位置信息来产生位置感知推荐，相比于传统推荐算法，PMCF 算法显著提高了推荐精度。

参 考 文 献

[1] 项亮，陈义，王益. 推荐系统实践 [M]. 河北：人民邮电出版社，2012：121-143，179-195

[2] Park M-H，et al. Location-Based Recommendation System Using Bayesian User's Preference Model in Mobile Devices [C]// Proceedings of the Ubiquitous Intelligence and Computing (UIC). Hong Kong，China，2007：1130-1139

[3] Netflix News and Info-Local Favorites [EB/OL]. [2013-09-10]. http://tinyurl.com/4qt8ujo

[4] Bao J，Chow C-Y，Mokbel M F，et al. Efficient evaluation of k-range nearest neighbor queries in road networks [C]// Proceedings of the International Conference on Mobile Data Management. Kansas City，MO，USA，2010：115-124

[5] Mouratidis K，Yiu M L，Papadias D，et al. Continuous nearest neighbor monitoring in road networks [C]// Proceedings of the Very Large Date Bases (VLDB). Seoul，Korea，2006：43-54

[6] Bruno N，Gravano L，Marian A. Evaluating Top-k Queries over Web-Accessible Databases [C]// Proceedings of the 18th International Conference on Data Engineering (ICDE). San Jose，California，USA，2002：369-380

[7] Venetis P，Gonzalez H，Jensen C S，et al. Hyper-Local，Directions-Based Ranking of Places [C]// Proceedings of the 27th International Conference on Data Engineering (ICDE). Hannover，Germany，2011：290-301

[8] Takeuchi Y，Sugimoto M. An Outdoor Recommendation System based on User Location History [C]// Proceedings of the Ubiquitous Intelligence and Computing (UIC). Wuhan and Three Gorges，China，2006：625-636

[9] Zheng V W，Zheng Y，Xie X，et al. Collaborative Location and Activity Recommendations with GPS History Data [C]// Proceedings of the 10th International Conference on World Wide Web. New York，USA，2010：1029-1038

[10] Ye M，Yin P，Lee W-C. Location Recommendation for Location-based Social Networks [C]// Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. San Jose，California，USA，2010：458-461

[11] Sarwar B，Karypis G，Konstan J，et al. Item-based Collaborative Filtering Recommendation Algorithms [C]// Proceedings of the 10th International Conference on World Wide Web. New York，USA，2001：285-295

[12] Fagin R，Lotem A，Naor M. Optimal Aggregation Algorithms for Middleware [C]// Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. New York，USA，2001：102-113

(上接第 315 页)

$GD(A-\{a_1,a_2\})=9/25$，

$Sig_{A-\{a_1,a_2\}}(a_1,a_2)$

$=GD(A-\{a_1,a_2\})-GD(A)=4/25$；

$GD(A-\{a_1,a_3\})=7/25$，

$Sig_{A-\{a_1,a_3\}}(a_1,a_3)$

$=GD(A-\{a_1,a_3\})-GD(A)=2/25$；

$GD(A-\{a_1,a_4\})=7/25$，

$Sig_{A-\{a_1,a_3\}}(a_1,a_4)$

$=GD(A-\{a_1,a_4\})-GD(A)=2/25$；

$GD(A-\{a_2,a_3\})=5/25$，

$Sig_{A-\{a_2,a_3\}}(a_2,a_3)$

$=GD(A-\{a_2,a_3\})-GD(A)=0$；

$GD(A-\{a_2,a_4\})=7/25$，$Sig_{A-\{a_2,a_4\}}(a_2,a_4)$

$=GD(A-\{a_2,a_4\})-GD(A)=2/25$；

$GD(A-\{a_3,a_4\})=5/25$，

$Sig_{A-\{a_3,a_4\}}(a_3,a_4)$

$=GD(A-\{a_3,a_4\})-GD(A)=0$；

由上面讨论得，系统的最小约简为 $red(A)=\{a_1,a_2\}$，次小约简为 $red(A)=\{a_1,a_3\}$ or $\{a_1,a_4\}$ or$\{a_2,a_4\}$。

结束语 本文主要针对没有约简核的信息系统，提出了基于粒计算的属性约简算法的改进。传统的基于粒计算的约简算法大多以约简核 $Core(A)$为基础来逐步计算属性对于核的重要度，从而确定出系统的约简集。然而有的信息系统可能没有约简核，因此，基于核的约简算法就失效了。针对这一情况，本文对约简算法进行了改进，改进后的算法既可以用于有约简核的系统也可以用于没有约简核的系统，并通过实验证明了该算法的可行性。

参 考 文 献

[1] 苗夺谦，王国胤，刘清，等. 粒计算：过去、现在与展望[M]. 北京：科学出版社，2007

[2] 王国胤，张清华，胡军. 粒计算研究综述[J]. 智能系统学报，2007，6(2)：8-26

[3] 徐伟华，刘士虎，张文修. 一般二元关系下信息系统知识的粒度描述[J]. 计算机工程与应用，2011，47(18)：40-44

[4] 胡峰，黄海，王国胤. 不完备信息系统的粒计算方法[J]. 小型微型计算机系统，2005，26(8)：1335-1339

[5] 刘清，刘群. 粒及粒计算在逻辑推理中的应用[J]. 计算机研究与发展，2004，41(4)：546-551

[6] 徐久成，史进玲，孙林. 一种基于相对粒度的决策表约 简算法[J]. 计算机科学，2009，36(3)：205-207

[7] 冯林，刘照鹏，方丹. 信息系统中粒计算模型及其属性约简方法[J]. 重庆邮电大学学报：自然科学版，2008，22(5)：652-655

[8] 赵敏，罗可，秦哲. 基于粒计算的属性约简算法[J]. 计算机工程与应用，2008，44(30)：157-159

[9] 陈玉明，苗夺谦，焦娜. 基于二进制粒与粒计算的属性约简[J]. 广西师范大学学报：自然科学版，2008，26(2)：81-84

[10] 王红霞，王志伟，程艳慧. 一种基于粒计算属性约简算法的改进及应用[J]. 微计算机信息，2010，26(5)：33-35

[11] 苗夺谦，范世栋. 知识的粒度计算及其应用[J]. 系统工程理论与实践，2002，22(1)：48-56

[12] 张文修，吴伟志，梁吉业，等. 粗糙集理论与方法[M]. 北京：科学出版社，2001

[13] 梁吉业，李德玉. 信息系统中的不确定性与知识获取[M]. 北京：科学出版社，2005

[14] Pawlak Z. Rough Sets：Theoretical Aspects of Reasoning about Data [M]. Dordrecht，Boston：Kluwer Academic Publishers，1991

[15] Liang J Y，Shi Z Z. The Information Entropy，Rough Entropy and Knowledge Granulation in Rough Set Theory [J]. International Journal of Uncertainty，Fuzziness and Knowledge-Based Systems，2004，12(1)：37-46