

基于 Hadoop 的旅游景点推荐的算法实现与应用

马腾腾¹, 朱庆华¹, 曹 菡², 沈 超³

(1. 南京大学 信息管理学院, 江苏 南京 210023;

2. 陕西师范大学 计算机科学学院, 陕西 西安 710119;

3. 南京邮电大学 经济与管理学院, 江苏 南京 210046)

摘 要:通过提高挖掘效率、增强算法扩展性,解决传统的推荐算法在旅游景点推荐方面响应时间长、推荐效率低,无法适应大数据挖掘需求的问题。对现有协同过滤推荐算法进行深入分析,选取适用于旅游景点推荐的 Slope One 算法和 Item-based 算法。将这两种算法高效结合,并基于 MapReduce 编程在 Hadoop 云平台上实现算法并行化,通过采集“旅评网”的真实旅游景点评分数据验证算法的有效性。通过测试真实的旅游景点评分数据,表明算法不仅提高了推荐的准确度,而且比传统的协同过滤算法具有更高的运行速度。实验结果较好地说明了该算法具有更高的挖掘性能和可扩展性,能够更好地适应旅游景点数据量大、数据矩阵稀疏的特性,满足旅游景点推荐高命中率和个性化的要求。

关键词:Hadoop; 旅游景点推荐; 协同过滤; Mahout; Item-based; Slope One

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2016)03-0047-06

doi:10.3969/j.issn.1673-629X.2016.03.012

Implementation and Application of Algorithm of Tourist Attractions Recommendation Based on Hadoop

MA Teng-teng¹, ZHU Qing-hua¹, CAO Han², SHEN Chao³

(1. School of Information Management, Nanjing University, Nanjing 210023, China;

2. School of Computer Science, Shannxi Normal University, Xi'an 710119, China;

3. School of Engineering and Management, Nanjing University, Nanjing 210046, China)

Abstract: The problems that the traditional tourist recommendation algorithm has long response and low efficiency in tourist recommendation, and have been unable to meet the mining needs of large amount of data can be solved by improving mining efficiency and enhancing scalability. It conducts a deep analysis for existing collaborative filtering algorithms in this paper and selects the Slope One algorithm and the Item-based algorithm suitable for tourist attractions recommendation to efficiently combine. Then the new algorithm is paralleled based on Hadoop framework, and the algorithm's validity is proved by the data collected from "www.ilvping.com". The faster speed and higher accuracy of the recommendation algorithm have been proved by the data collected from "www.ilvping.com". Experimental results show that the new algorithm has a better performance and scalability, which can be better to solve the problems of big data and sparse matrix, and meet the requirement of high percentage shot and personalization.

Key words: Hadoop; tourist attractions recommendation; collaborative filtering; Mahout; Item-based; Slope One

1 概 述

旅游业是国内经济支柱产业之一,近年来国内旅游行业持续升温。在旅游领域,随着信息技术发展,互联网信息飞速膨胀,使得旅游者对信息的使用效率大大降低,因此关于旅游服务的各种推荐应运而生。

目前关于旅游的推荐有景点推荐、线路推荐和酒

店推荐等^[1]。文献[2]设计和实现了一个基于混合模式的简单的个性化旅游推荐原型系统;文献[3]设计了一种本体和地理信息系统相结合的推荐系统;文献[4]使用协同过滤算法对从在线旅游网站上提取的游客对桂林旅游景点的评价数据进行旅游景点推荐。在推荐算法上,普遍使用的推荐算法有基于协同过滤的

收稿日期:2015-06-21

修回日期:2015-09-23

网络出版时间:2016-02-18

基金项目:国家自然科学基金资助项目(71473114)

作者简介:马腾腾(1992-),女,硕士研究生,研究方向为互联网用户行为分析;朱庆华,教授,博士生导师,研究方向为互联网用户行为分析;曹菡,教授,博士生导师,研究方向为并行计算与大数据处理、GIS与智慧旅游。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160218.1634.060.html>

推荐、基于内容的推荐和基于模型的推荐等。其中基于协同过滤的推荐算法在个性化系统中应用最广^[5]。同时,针对协同过滤的推荐算法的研究成果比较多,如 MovieLens^[6]、GroupLens^[7]、视频推荐系统^[8]以及 Ringo^[9],这些应用都在互联网上拥有广泛的用户。在商业方面,一些知名的网站已经成功运用协同过滤技术,如 Amazon.com、Launch.com、MovieFinder.com 和 CDNow.com。近几年,研究学者也对协同过滤算法进行了很多优化与改进。Sarwar B 等^[10]研究了基于项目的协同过滤推荐算法,并对余弦相似度算法进行了优化。Lemire D 等^[11]提出了 Slope One 算法,较好地解决了数据稀疏性和冷启动问题。

信息化时代在为旅游者提供便捷的同时也带来了挑战。虽然潜在旅游者可以借助互联网搜集到大量的信息,但由于信息量大且参差不齐,信息搜集者需要耗费大量的时间去筛选,且仍不能保证最终信息搜集的准确性。这种情况大大增加了旅游者制定出行计划的难度,增加了旅游者的出行成本,降低了对景点的满意度。随着推荐系统得到广泛的认可,现有的推荐系统也面临着关键性挑战,如何提供高质量的推荐结果,如何能快速为数以万计的用户做出高质量的推荐,如何能在数据极度稀疏的情况下满足高命中率和个性化的需求,这些都是亟待解决的问题。

文中结合旅游消费行为的特点与大数据的特性,探索一种针对旅游景点推荐的协同过滤算法,提出一种基于项目的协同过滤算法和 Slope One 算法相结合的解决方法,从而进一步提高推荐的速度和精度,同时使用 Apache Mahout 项目在 Hadoop 平台上将其实现。

2 推荐算法的优化

文中采用 Slope One 与 Item-based 的结合算法,能有效针对旅游景点进行推荐。

2.1 Slope One 算法

Daniel Lemire 和 Anna Maclachlan 在 2005 年发表的论文中提出 Slope One 算法^[11]。Slope One 算法是对基于项目的协同过滤算法的简化和改进。该算法的实现过程简单而高效,而且与其他费时复杂的算法相比,其精度不相上下^[12]。

Slope One 算法的原理是根据各个用户对各项目的偏好平均分差来对未评分的项目进行评分预测。计算平均分差的公式为:

$$\text{dev}_{j,i} = \sum_{u \in s_{j,i}(x)} \frac{u_j - u_i}{\text{card}(s_{j,i}(x))} \quad (1)$$

其中: s 表示同时对景点 i 和 j 给予了评分的用户集合; card 表示集合中所包含的评分数量。

根据式(1)就可以获得某特定用户对未评分景点

的预测评分结果。当把所有可能的预测评分平均起来,就得到式(2):

$$P(u)_j = \frac{1}{\text{card}(R_j)} \sum_{i \in R_j} (\text{dev}_{j,i} + u_i) \quad (2)$$

其中: $P(u)_j$ 表示用户对景点 j 的预测评价; R_j 表示所有已经被用户 u 给予评价的项目集合。

Slope One 算法适用于旅游景点推荐的原因有:

第一,与其他协同过滤算法比较而言,Slope One 算法无需计算景点或者用户的相似度,直接根据用户的已有评分进行预估,原理简单,计算速度快,能在海量旅游数据中快速得出推荐结果,提高效率。

第二,Slope One 算法对评分记录很少的用户也能得出推荐结果,有效地解决了稀疏矩阵问题。这对实际运作中推荐系统能否在稀疏矩阵的情况下进行有效推荐具有重要意义。

2.2 基于项目的推荐算法

基于项目的协同过滤(Item-based collaborative filtering)算法是目前业界应用最多的算法,主要分为两步:

(1)计算物品与物品之间的相似度。

Pearson 相似度算法:设 U 为对项目 i 和项目 j 都评过分的用户集合,且 $U_{ij} = U_i \cap U_j$,则 $\text{Sim}(i,j)$ 为:

$$\text{Sim}(i,j) = \frac{\sum_{k \in U_{ij}} (R_{k,i} - \bar{R}_i)(R_{k,j} - \bar{R}_j)}{\sqrt{\sum_{k \in U_{ij}} (R_{k,i} - \bar{R}_i)^2} \sqrt{\sum_{k \in U_{ij}} (R_{k,j} - \bar{R}_j)^2}} \quad (3)$$

其中: $R_{k,j}$ 为用户 k 对项目 i 的评分; \bar{R}_i 、 \bar{R}_j 分别为对项目 i 和 j 的平均评分。

(2)根据物品之间的相似度和用户的行为记录,使用权重求和的方法来计算预测评分,给用户推荐相应物品。目标用户 u 对项目 i 的预测评分 $P_{u,i}$ 为:

$$P_{u,i} = \frac{\sum_{\text{similar-item},n} (S_{i,n} * R_{u,n})}{\sum_{\text{similar-item},n} (|S_{i,n}|)} \quad (4)$$

其中: $R_{u,n}$ 表示用户 u 对相似项目集中项目 n 的评分; $S_{i,n}$ 表示项目 i 与项目 n 的相似程度。

选择基于项目的协同过滤算法作为景点推荐算法的原因是每个人喜欢的景点大多局限于一类或两类中,且具有一定的相似性。而基于项目的协同过滤算法即是通过计算景点之间的相似度,将与用户喜爱的景点相似的推荐给用户,符合旅游行为的特征。

2.3 推荐算法的改进

人们喜爱的景点大多仅限于特定类型。对此,基于项目的协同过滤算法完全符合相似性这一旅游行为特点。该算法能够根据其他用户对景点的偏好,发现与当前用户喜好的景点相类似的景点,为当前用户进

行推荐。然而,该算法不能解决稀疏矩阵的问题,这在实际的推荐系统中反映为不能为新用户进行推荐,大大影响了算法的整体性。

Slope One 算法则不同,即使只有一条数据该算法也能进行推荐,且运算速度快。如果将这两种算法相结合,既能根据旅游数据的特点进行有效推荐,又能解决数据稀疏性问题。

文中根据基于项目协同过滤算法和 Slope One 推荐算法的优缺点,将两种算法相结合,共同构成旅游景点的推荐算法,并利用真实的旅游数据进行测算。统计后发现在用户记录数低于 3 的情况下,基于用户协同过滤算法由于数据量太小不能得出推荐结果,因此选择记录数“3”作为算法选择的临界点。如果记录数大于等于 3,那么就通过 Item-based 算法得到推荐结果,否则采用 Slope One 算法获得推荐结果。

该方法的优点是将两种算法相结合,进一步提高了旅游景点推荐的准确性,而且解决了数据稀疏性问题。

3 推荐系统的实现

3.1 总体架构设计

将推荐系统在云计算平台上实现,不仅可以解决超大规模数据集的存储、处理难题,而且增强了系统的可扩展性。文中在 Hadoop 云平台上研究旅游景点推荐算法,其框架如图 1 所示。

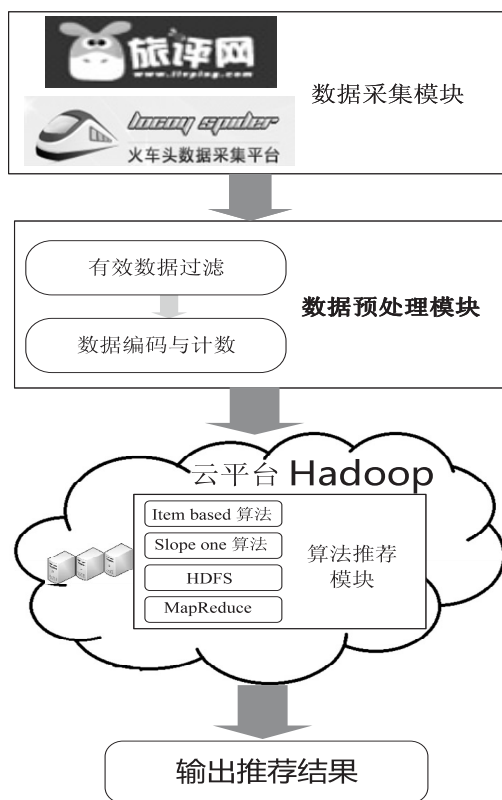


图 1 景点推荐系统框架

该系统主要包括数据采集模块、数据预处理模块和算法推荐模块三部分。

数据采集模块:利用“火车头”软件从“旅行网”上爬取所需的数据,即用户对景点的评分集。文中使用的所有数据均来自于正规的旅游网站,真实可靠,确保了算法与旅游景点推荐的切合性;

数据预处理模块:将采集来的数据进行编码,转化为 CSV 文件格式,以便算法程序能够识别;

算法推荐模块:以 Hadoop 平台为基础,将 MapReduce 并行化,推荐算法后台程序生成推荐模型,根据采集来的真实数据计算推荐结果^[13-14]。

3.2 数据的获取

3.2.1 采集网站

文中选择“旅评网”作为数据采集网站。旅评网是国内领先的在线旅游景点信息平台,同时也是旅行者对景点点评和经验分享的社区。鉴于旅评网在驴友中有较好的口碑,是分享旅游心得和景点评分的权威平台,所以选择该网站作为数据爬取网站。

文中所需要的数据格式为“用户+景点+评分”,因此对旅评网上各个景点的名称和评论该景点的用户名及其对该景点的评分进行采集。

3.2.2 采集工具

文中使用火车头采集器(LocoySpider)进行旅游数据的采集。该软件是一款功能齐全且易于上手的专业采集软件,具有内容采集和数据导入、导出功能。利用系统内置标签,将采集到的数据对应表的字段导出到本地的 Access 数据库中。

3.2.3 采集过程

数据采集采用“前后截取”的方法,根据网页源代码,确定所需内容所在的位置并截取。文中只截取三列信息,分别为:用户、景点和评分。

3.2.4 数据预处理

采集来的景点名称、用户名都是汉字,在 Hadoop 下 Mahout 不能识别,因此必须对景点名称和用户名进行编码,用户编码为 1 ~ 1404,景点编码为 10001 ~ 10200。

整理后得到评分数据 5 144 条,1 404 个用户,景点 200 个。

3.3 Hadoop 平台上景点推荐算法的实现

3.3.1 Hadoop 平台的搭建

Hadoop 的运行模式主要有三种:单机模式、伪分布式和完全分布式。文中搭建的是伪分布式 Hadoop 平台。

(1) 硬件配置。

处理器: Intel Core i5 - 2400 (3.10 GHz 四核 64 bit);

内存(RAM):4 GB;
 硬盘:500 GB(7200 转);
 主板:HP 1497(Inter Q65 芯片组)。

(2) 软件环境。

操作系统:Ubuntu 12.10;

JDK 版本:1.7.0_04。

3.3.2 用户记录的计数

文中需要对用户评分数据进行计数,根据记录数将用户分类选择合适的算法。由于数据量庞大,不适宜采用耗时耗力的人工计数。Hadoop 平台具有运算速度快、稳定性高的特点,因此文中利用搭建好的 Hadoop 平台,将用户数据上传至 HDFS,调用 WordCount 程序,计算出每个用户的评分数,以便进行下一步的算法选择。

3.3.3 算法的 Map/Reduce 化

在 Hadoop 环境下,Slope One 推荐算法的整个过程如下:

- (1) 根据用户 id 找出用户没有偏好过的集合作为被推荐物品,并找出用户偏好过的物品及其评分。
- (2) 对每一个被推荐物品列表中的物品 i,根据用户偏好过的物品列表中物品对 i 计算预测评分。
- (3) 把所有的推荐物品及其预测评分组成一个列表并排序。没有评分的可以省去。
- (4) 对推荐结果进行处理。

图 2 是描述 Slope One 算法计算用户对被推荐物品的预测评分的伪代码。其主要过程如下:

```
Map (user_id, value){//key: userid, value: item_i, item_j
    (item_i,item_j) = value.split(",");//从 value 中获取两个物品 id
    Ratings_i = getrating(item_i); //Ratings_1 是物品 i 的评分向量
    Ratings_j = getrating(item_j); //Ratings_2 是物品 j 的评分向量
    For (r1,r2) in (Ratings_i 并 Ratings_j){这里的并值的是有一个用户同时 d 对这两个物品评分
        count++; //计算同时评分的个数
        T += r2-r1; //计算公式 2-11 中的  $\Sigma(u_j-u_i)$ 
    }
    If(count==0) //如果不存在同时评分的领个物品那么就不进行输出。
        Outputkeyvalue(null);
    R_ui = rating(user_id,item_i); //R_ui 是 user_id 对 item_i 的评分。
    T = T/count; //计算公式 2-10 中单个分子。
    Outputkeyvalue(item_j,count+" "+T); //value 中同时把 count 传递给 reducer
}

Reducer(item_j,values){
    For each v in values
        (count,T) = v.split(",");//分割出 count 和 T
        Pred_T += T;
        pred_c += count;
    Predictive = pred_T/pred_c; //根据公式 2-10 计算出预测评分。
    Outputkeyvalue(item_j,Predictive);
}
```

图 2 Slope One 的 MapReduce 化

Map 的输入是以 userId 为键值,根据步骤 2 组合的 itemId 为 value。Map 输出的结果是以要预测的物品 j 为 key 值,以同时评分个数 count 以及计算结果的字符串连接为 value。在 Reducer 之前这些结果会被 MapReduce 框架合并在一起交给 Reducer 计算。Re-

ducer 经过计算,其结果即为用户对物品 j 的预测评分。

Item-based 实现的主要流程就在 Mahout 包的 RecommenderJob 类中。所有的步骤均包含在这个类的 run 方法中。计算过程共分为 10 步,其中相似度的计算被拆分成 3 个 job。图 3 为详细的步骤分析。

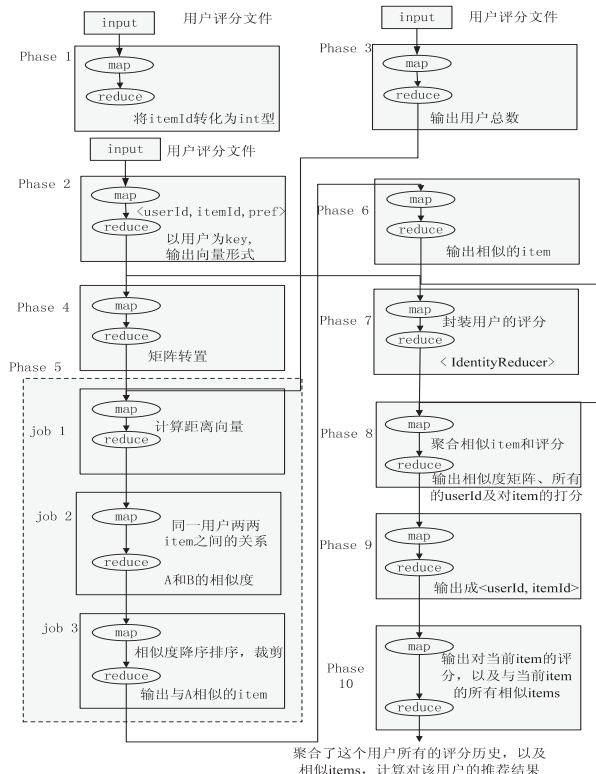


图 3 Item-based 的 MapReduce 化

至此,已经形成了推荐结果,Item-based 推荐算法完成。

4 结果分析及讨论

对任意一个用户,推荐系统会按照推荐算法模块中的设定条件,选择基于项目协同过滤算法或者 Slope One 算法进行计算,得出一组推荐景点的 ID 和用户对该景点的预估评分(文中设定为每位用户推荐 10 个景点),并按预测评分降序排序。相应内容如图 4 所示。

```
1  [10113:5.6666665,10163:5.6666665,10012:5.6,10096:5.5,10114:5.5,
2  [10128:5.0,10199:5.0,10089:4.5,10016:4.5,10161:4.5,10087:4.5,100
3  [10082:6.0,10159:5.5,10081:5.5,10023:5.5,10045:5.5,10038:5.0,10
4  [10154:6.0,10028:6.0,10082:6.0,10071:5.75,10187:5.6666665,10155
5  [10073:7.0,10051:6.5,10081:6.0,10046:6.0,10083:6.0,10158:6.0,100
6  [10096:4.6666665,10128:4.5,10113:4.5,10091:4.5,10072:4.5,10012:
7  [10082:6.0,10012:6.0,10187:5.6666665,10023:5.5,10046:5.5,10080:
8  [10128:6.0,10095:5.692811,10096:5.690847,10077:5.6666665,1005
9  [10128:5.0,10091:5.0,10058:5.0,10113:5.0,10052:4.75,10085:4.75,1
10 [10102:5.0,10069:5.0,10006:4.75,10087:4.5,10052:4.428571,10128:
```

图 4 推荐结果示例

结果的输出格式为:

<用户代码 [推荐景点 1:预测评分 1,……,推荐景点 10:预测评分 10]>

每行中的第一个数字是用户代码,中括号里的是推荐的景点代码及其对应的预测评分,按评分由高到低排列。该算法为每个用户推荐 10 个景点,以逗号间隔。如图 4 所示,为 1 号用户推荐 10113 号景点,且该用户对 10113 号景点的预估评分为 5.67 分。

为了对比各个算法的运算速度以及对旅游数据的适用程度,在算法实现后,笔者从“旅评网”采集三组大小不同的旅游数据,调试和运行分布式协同过滤算法。

此次采集的旅游数据共有 5 144 条评分记录,如表 1 所示。

表 1 数据采集统计表

用户数量	景点数量	评分记录数量
534	50	1 680
895	120	3 027
1404	200	5 144

在算法的精确度方面,文中采用平均绝对误差 (MAE) 作为评价指标。它是协同过滤算法中统计精度的常用评价标准,其计算公式为:

$$MAE = \frac{\sum_{i=1}^N |R_{u,i} - R'_{u,i}|}{N}$$

(5)

其中: $R_{u,i}$ 表示用户 u 对项目 i 的实际评分; $R'_{u,i}$ 表示推荐算法的预测分数; N 表示项目数量。

MAE 越小,表示精度越高。在评分记录数不同的情况下,各个算法的 MAE 值如图 5 所示。

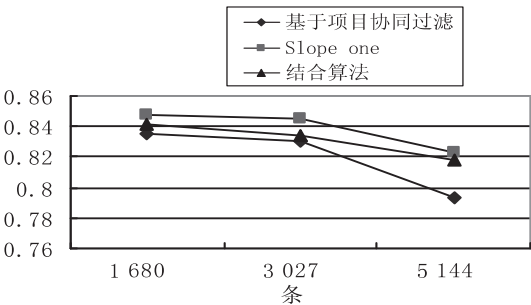


图 5 MAE 比较

可以看出,随着数据量逐渐增大,各个算法的精度都有所提高。在数据量相同的情况下,基于项目协同过滤算法的精度始终高于 Slope One 算法,结合算法的精度位于二者之间。

在算法的耗时方面,文中依然取以上三组数据作为实验数据,测算基于用户协同过滤算法、Slope One 算法以及结合算法对数据处理的响应时间,结果如图 6 所示。

可以看出,基于项目协同过滤算法花费的时间要比 Slope One 算法多。原因是基于项目协同过滤需要计算景点之间的相似度,然后将相似度加权计算得到

最终推荐结果,而 Slope One 算法无需计算景点或者用户的相似度,直接根据用户已有数据进行预估评分。结合算法的运算速度介于两者之间。

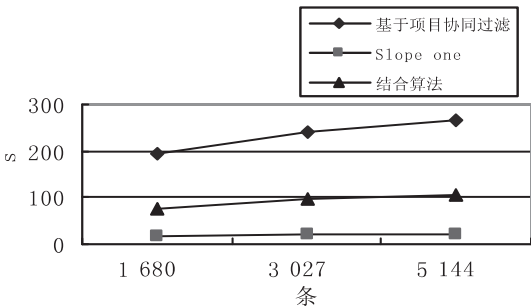


图 6 响应时间比较

通过以上实验结果得出,基于项目协同过滤算法在精度上优于 Slope One 算法,然而通过分析采集来的旅游数据发现,人们受各种因素的限制,去过的旅游景点数较为有限,因此评分数据矩阵相比于电影评分、图书评分等较为稀疏。这一点决定了基于项目协同过滤算法在旅游景点推荐上具有一定的局限性。虽然 Slope One 算法在精度上不及基于项目协同过滤算法,却能有效解决数据稀疏性问题,且运算速度较快。结合算法在精度和运算时间上均介于两种算法之间,发挥了两种算法的优势,因此十分适用于矩阵稀疏、数据量较大的旅游景点推荐。

从算法的应用效果来看,将 Slope One 与基于项目的协同过滤算法相结合,扬长避短,既发挥了 Slope One 算法运算速度快的优势,又利用基于项目协同过滤算法提高了推荐结果的准确性;其次,在用户记录数很少的情况下选用 Slope One 算法,有效地解决了数据稀疏性问题,使系统即使在数据矩阵十分稀疏的情况下也能进行推荐;并且,使用的基于项目协同过滤算法和 Slope One 算法能贴切地迎合旅游消费者的动机和行为,提高用户的体验,避免了用户需要花费大量的时间才能找到自己感兴趣的景点的情况,具有一定的针对性。

从算法的运行效果来看,系统能准确地按照当前用户的喜好信息以及相似用户的喜好信息推荐旅游景点。系统通过算法计算出的各个景点的预测值,并将推荐结果的预测值由高到低进行排序,将排名前 10 的景点推荐给用户。同时,通过多次测试表明,相同用户在喜好信息没有发生变化的前提下,该推荐系统的推荐结果保持不变,表现出较好的算法稳定性。因此在推荐结果方面,该景点推荐系统能根据用户对景点的已有评分较为准确地预测对未评分景点的喜好程度,并依据预测评分进行景点推荐。

在技术创新应用方面,利用 Hadoop 云平台进行服务推荐不仅可以提高数据存储与处理能力,而且可以

增强系统的可扩展性。

5 结束语

文中提出了 Item-based 协同过滤算法与 Slope One 推荐算法相结合的解决方案。采集真实旅游景点评分数据,将推荐系统在 Hadoop 云平台上加以实现,具有一定的实际应用价值。

同时,算法仍存在一些不足和需要进一步研究的方面:

(1)文中在云平台上实现了 Item-based 和 Slope One 相结合的算法,并没有真正实现推荐系统,仅对算法进行了实现与应用。今后可以考虑开发完整的旅游景点推荐系统。

(2)文中主要根据用户对景点的已有评分预测用户的偏好,考虑的因素不够全面。在今后的研究中,可以综合多个影响评分的因素,如在算法中加入用户的社交关系和情感因素等参数。

参考文献:

- [1] 汪英姿,马 慰,徐守坤.一种基于本体的旅游资源二次推荐方法[J].情报科学,2012,30(12):1866-1871.
- [2] 石 静.基于混合模式的个性化推荐系统的应用研究[D].武汉:武汉理工大学,2010.
- [3] 胡纳纳,李琳琳,武 尚.个性化的旅游推荐系统[J].信息技术,2013(2):135-139.
- [4] 侯新华,文益民.基于协同过滤的旅游景点推荐[J].计算机技术与自动化,2013,31(4):116-119.
- [5] 奉国和,梁晓婷.协同过滤推荐研究综述[J].图书情报工作,2011,55(16):126-130.
- [6] Bradley N M, Albert I, Shyong K L, et al. MovieLens un-

plugged; experiences with an occasionally connected recommender system[C]//Proceedings of the 8th international conference on intelligent user interfaces. New York, USA: [s. n.], 2003:263-266.

- [7] Konstan J A, Miller B N, Maltz D, et al. GroupLens: applying collaborative filtering to Usenet news[J]. Communications of the ACM, 1997, 40(3): 77-87.
- [8] Hill W, Stead L, Rosenstein M, et al. Recommending and evaluating choices in a virtual community of use[C]//Proceedings of the conference on human factors in computing systems. Denver, USA: ACM, 1995: 194-201.
- [9] Shardanand U, Maes P. Social information filtering: algorithms for automating "word of mouth" [C]//Proceedings of the conference on human factors in computing systems. Denver, USA: ACM, 1995: 210-217.
- [10] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms [C]//Proceedings of the 10th international world wide web conference. Hong Kong, China: IW3C2, 2001: 285-295.
- [11] Lemire D, MacLachlan A. Slope one predictors for online rating-based collaborative filtering[C]//Proceedings of SIAM data mining. Newport Beach, California: Compensation Committee, 2005.
- [12] 王 毅,楼恒越.一种改进的 Slope One 协同过滤算法[J].计算机科学,2011,38(10A):192-194.
- [13] Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters[J]. Communications of the ACM, 2008, 51(1): 107-113.
- [14] Shvachko K, Kuang H, Radia S, et al. The Hadoop distributed file system[C]//Proceedings of the 26th symposium on mass storage systems and technologies. Incline Village, USA: IEEE, 2010: 1-10.

(上接第 46 页)

- [3] 王 帆,赵春晖,张 志.基于克隆选择算法的人脸特征选择[J].应用科技,2009,36(3):11-14.
- [4] 刘 琼,彭光正.一种改进的 AdaBoost 人脸检测算法[J].计算机应用与软件,2011,28(6):265-268.
- [5] 伍世虔.红外人脸识别[J].西安邮电大学学报,2013,18(1):11-21.
- [6] 王忠民,曹 栋.基于蚁群算法的行为识别特征优选方法[J].西安邮电大学学报,2014,19(1):73-77.
- [7] 侯梦华.基于机器视觉的实时目标检测系统研究[D].广州:广东工业大学,2008.
- [8] 陆振杰,宋 进.单幅数字图像多尺度空间下的场景深度估计[J].计算机技术与发展,2013,23(1):51-53.
- [9] 于新辉,张 建,李伟涛.基于生命周期分析信息安全管理体

1751-1754.

- [11] 衡 霞,王忠民.基于手机加速度传感器的人体行为识别[J].西安邮电大学学报,2014,19(6):76-79.
- [12] 吴 青,赵 雄.一类新样条光滑支持向量机[J].西安邮电大学学报,2013,18(6):68-74.
- [13] 何世民.基于 Adaboost 算法的人脸检测及其在 DSP 平台上的移植[D].哈尔滨:哈尔滨理工大学,2010.
- [14] Papageorgiou C, Poggio T. A trainable system for object detection[J]. International Journal of Computer Vision, 2000, 38(1):15-33.
- [15] Huang C, Ai H, Li Y, et al. High performance rotation invariant multi-view face detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(4): 671-686.
- [16] 李瑞淇.基于肤色和改进的 Adaboost 人脸检测算法研究[D].西安:西安电子科技大学,2014.