

引用格式: 彭浩, 佃松宜. 基于改进 FP-growth 的用户兴趣推荐算法的设计与实现[J]. 微电子学与计算机, 2020, 37(2): 8-13. [PENG H, DIAN S Y. Design and implementation of a user interest recommender algorithm based on improved FP-growth [J]. Microelectronics & Computer, 2020, 37(2): 8-13.]

# 基于改进 FP-growth 的用户兴趣推荐算法的设计与实现

彭 浩, 佃松宜

(四川大学 电气信息学院, 四川 成都 610065)

**摘 要:** 针对一类只有用户浏览频次数据且没有项目评分的推荐系统, 提出一种基于 FP-Growth(频繁模式增长)改进的关联规则挖掘算法, 实现用户兴趣推荐. 针对传统的 FP-growth 进行如下改进: 首先, 归一化用户浏览频率, 以反映用户的兴趣, 并添加遗忘因子来跟踪用户兴趣迁移. 其次, 采用动态支持度过滤非兴趣项目, 以提高推荐结果的准确性. 此外, 在 FP-tree 中引入项目活跃度作为冷启动问题的可行方案. 最后收集 762 个用户在企业操作系统(EOS)区块链中访问智能合约的记录来测试改进的算法. 实验结果表明, 该算法在用户兴趣推荐方面表现令人满意.

**关键词:** 推荐系统; FP-growth; 关联规则挖掘; 冷启动

**中图分类号:** TP391. 3

**文献标识码:** A

**文章编号:** 1000—7180(2020)02—0008—06

## Design and implementation of a user interest recommender algorithm based on improved FP-growth

PENG Hao, DIAN Song-yi

(College of Electrical Engineering and Information Technology, Sichuan University, Chengdu 610000, China)

**Abstract:** In this paper, a novel algorithm of mining association rules based on improved FP-growth which focuses on precise user interest recommendation, is proposed for the system without user ratings but a large amount of browsing data. The improvements based on traditional FP-growth can be particularized in the following: First of all, the frequency of user browsing is normalized and multiplied by forgetting function in order to reflect the current interest of users. Besides, dynamic support is adopted to filter out non-interest items and enhance the recommending accuracy. Moreover, activity of item (AOI) is employed as a feasible scheme in the FP-tree to solve the cold start problem. Then, the improved algorithm is verified by the browsing data of the smart contracts in the Enterprise Operation System (EOS) blockchain. The experimental results show that the improved algorithm performs satisfactory in the user interest recommendation.

**Key words:** recommender system; FP-growth; association rule mining; cold start

### 1 引言

互联网服务的发展为人们的日常交流和兴趣发现提供了前所未有的便利<sup>[1]</sup>. 然而, 信息过载增加了在众多数据源中选择最佳项目的困难<sup>[2]</sup>. 推荐系统

作为解决用户个性化需求和信息超载之间矛盾的解决方案, 得到了广泛的研究和开发<sup>[3-5]</sup>.

本文的研究重点是针对没有用户评级的系统挖掘用户兴趣. 主流的基于矩阵的协同过滤(CF)算法存在矩阵稀疏性的问题, 文献[2][3][6][7]关于改

进的矩阵分解 CF 算法可以缓解矩阵稀疏性带来的计算代价,但是对于用户数据更新会带来大量计算<sup>[8]</sup>,同时对于非兴趣项目的处理能力不理想<sup>[9]</sup>. FP-growth 算法的理想挖掘效率和对矩阵稀疏性的不敏感性促使我们采用基于改进 FP-growth 的 CF 来解决研究.

传统 FP-growth 算法因其算法步骤简单、内存要求低、运行效率高等优点在关联规则挖掘中占据重要地位<sup>[10-11]</sup>. 过去十年里,由于硬件设备的限制,关于 FP-growth 算法的研究主要为了提高算法的效率和降低计算复杂度. 文献[10]提出了一种新颖的 FP 阵列技术,以减少 CPU 遍历 FP 树的时间. 为了提高挖掘效率,文献[11],同时构建增量 HUP 词典图形树, IHUP 事务频率树和 IHUP 事务加利用树,协同工作以进行有效的数据挖掘. 近十年来关于 FP-growth 算法的研究,更关注系统应用. 文献[12]通过修剪树来克服维度的爆炸并支持快速在线推荐. 文献[13]巧妙地应用 FP-growth 算法生成一组候选模糊关联规则,以提高分类精度.

本文的主要工作为对文献[14][15]提出 FP-growth 算法进行改进,使其能够根据用户访问记录反映用户兴趣特征,引进动态支持度实现精确推荐,并提供冷启动解决方案.

2 提出的改进

传统 FP-growth 算法根据用户数据挖掘项目频繁项集,再通过项目频繁项集建立用户与项目之间的联系,以此实现关联规则挖掘,并用于推荐系统. 本节根据传统 FP-growth 算法步骤提出相应的改进和扩展,以解决本文的研究问题.

2.1 相关说明

为了方便接下来的叙述和推导,这里给出研究问题的数据特征和相关符号的描述.

本文研究的用户数据经统计稀疏度为 96%,为极度稀疏系统.  $I = \{i_1, i_2, \dots, i_n\}$  表示项目的集合,  $U = \{u_1, u_2, \dots, u_m\}$  表示用户的集合,其中  $n$  表示总项目数,  $m$  表示总用户量. 表 1 中列出了关键的符号,以便下文使用. 随机截取一个用户从 2018.08 — 2018.12 期间的浏览记录如表 2 所示,从表 2 中可以看出用户浏览记录特点以及数据稀疏性.

2.2 用户兴趣反映

针对没有评分的推荐系统,用户数据为针对不同项目的访问频次记录. 传统 FP-growth 算法挖掘用户兴趣,只能得出项目  $i$  是否被用户  $u$  访问,而无

法记录  $i$  被  $u$  访问多少次. 因此考虑归一化访问频次来反映用户对项目的喜爱程度. 用户  $u$  对于项目  $i$  的兴趣值由式(1) 计算:

表 1 符号说明

| 符号        | 描述              |
|-----------|-----------------|
| $I$       | 项目的集合           |
| $i$       | 单个项目, $i \in I$ |
| $U$       | 用户的集合           |
| $u$       | 单个用户, $u \in U$ |
| $S$       | 支持度             |
| $\lambda$ | 遗忘因子            |
| $f$       | 频次              |
| $t$       | 数据产生时间          |
| $p$       | 兴趣率             |
| AOI       | 项目活跃度           |

表 2 数据类型和特点

| 月份  | $c$ | $d$ | $f$  | $g$ | $h$ | $m$ | $q$ | $t$ | $z$ |
|-----|-----|-----|------|-----|-----|-----|-----|-----|-----|
| 八月  | 875 | 22  |      |     | 2   | 13  |     | 1   |     |
| 九月  | 196 |     | 14   |     |     | 51  | 3   |     | 1   |
| 十月  | 44  | 18  | 389  |     |     |     |     |     |     |
| 十一月 | 56  |     | 2578 | 11  |     |     |     |     |     |
| 十二月 | 13  |     | 5188 | 34  |     |     |     |     | 2   |

$$p_{u,k} = \frac{f_{u,k} - \min(f_{u,i})}{\max(f_{u,i}) - \min(f_{u,i})}, i \in I \quad (1)$$

式中,  $f_{u,i}$  表示用户  $u$  对项目  $i$  的访问频次. 式(1) 虽然将兴趣值反映到 (0,1) 区间,但是无法反映出用户兴趣随时间发生的改变. 因此,引入遗忘函数(2) 来反映用户随时间变化的兴趣值.

$$f(t) = \exp(-\lambda \cdot \Delta t) \quad (2)$$

$\lambda$  作为遗忘因子控制遗忘的速度. 根据经验值,让数据的权重在 90 天之后衰减为 0.5<sup>[16]</sup>,带入式(2) 可得: $\lambda = 0.007\ 7$ ,解得  $0.5 = \exp(-\lambda \cdot 90)$ . 结合式(1) 和式(2) 可得:

$$p_{u,k} = \frac{\sum_{f_{u,k}} \exp(-0.007\ 7 \cdot \Delta t) - \sum_{\min(f_{u,i})} \exp(-0.007\ 7 \cdot \Delta t)}{\sum_{\max(f_{u,i})} \exp(-0.007\ 7 \cdot \Delta t) - \sum_{\min(f_{u,i})} \exp(-0.007\ 7 \cdot \Delta t)} \quad (3)$$

式中,分子表示加入遗忘因子之后用户  $u$  关于项目  $i$  的频次(不一定为整数). 式(3) 结合了归一化和遗忘函数,将用户的数据统一了量纲,最终能够反映用户的实时兴趣. 从数据库中随机读取一个用户的数据,用户的兴趣只反映过程如图 1 所示.

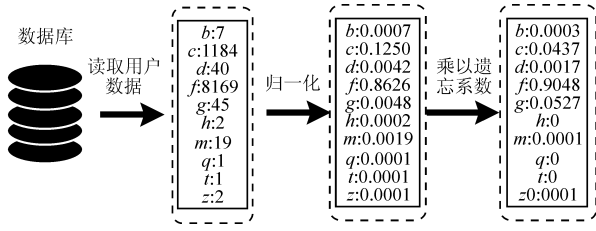


图1 压缩数据表现用户兴趣值

### 2.3 动态支持度

传统 FP-growth 算法在建立项头表时,通常根据总用户量设置支持度.当取总用户量的 2% 时<sup>[17]</sup>,支持度计算方式为:

$$S_k = \text{size}(U) \cdot 2\% \quad (4)$$

式(4)只能根据项目被调用的用户量与总用户量的占比来选择项目,从而忽略项目自身的特殊性.比如,当一个广告项目推送给大量的用户,这个项目会被认为被大多数人喜欢,这显然是错的.因此希望通过动态支持度保留兴趣项目而滤除非兴趣项目.考虑式(3)已经计算出每个项目在用户中的访问频次占比,如果将这个占比作为用户兴趣因子乘以当前支持度,那么兴趣项目的支持度就会明显高于非兴趣项目.考虑到求解每个项目的总占比来选择项目,从而达到保留兴趣项目滤除非兴趣项目.所以有:

$$S'_k = \text{size}(U) \cdot \sum_u \frac{p_{u,k}}{U_k}$$

$$= \text{size}(U) \cdot \sum_u \frac{\sum_{f_{u,k}} e^{-0.0077 \cdot \Delta f} - \sum_{\min(f_{u,i})} e^{-0.0077 \cdot \Delta f}}{\sum_{\max(f_{u,i})} e^{-0.0077 \cdot \Delta f} - \sum_{\min(f_{u,i})} e^{-0.0077 \cdot \Delta f}} \quad (5)$$

式中,  $U_k$  表示浏览过  $k$  项目的用户.此刻再建立项头表,比较式(5)和式(4)的大小:当  $S'_k \geq S_k$  时,视为该项目达到阈值要求;当  $S'_k \leq S_k$  时,被滤除.这样一来,被多数人喜爱的项目就更容易被保留到项头表中,而那些非兴趣项目就越容易被滤除.不仅如此,结合式(3),遗忘因子的作用也能反映到项头表中,过去被大多数人喜欢的项目,如果现在几乎没人使用,那么也很难出现推荐候选集中,最终得以实现精确的用户兴趣推荐.

### 2.4 IFP-tree

尽管本文对提高推荐精度做出了努力,但冷启动问题却没有得到有效解决.而幸运的是,通过上面的改进,为我们改善冷启动提供基础.下面我们对 FP-tree 进行扩展,针对传统 FP-tree 简单的结构,

引入项目活跃度来表示项目受欢迎的程度,并将该信息加入每个树枝节点.定义项目活跃度如下.

定义1:我们称项目活跃度为每个项目的节点数和利息总值的乘积.计算方法如下:

$$AOI(k) = \text{size}(U_k) \cdot \sum_u p_{u,k} \quad (6)$$

右边求和因式表示项目  $k$  的一个兴趣总值.从式(6)可以看出活跃度越高的越靠近树根节点,但也可能出现有些项目因为兴趣总值高而排名靠前.此外设置的活跃度并不影响树的结构,只是让每个项目在树中表达更多的信息.为方便区分,称新的树为 IFP-tree.整个过程如图2表示.

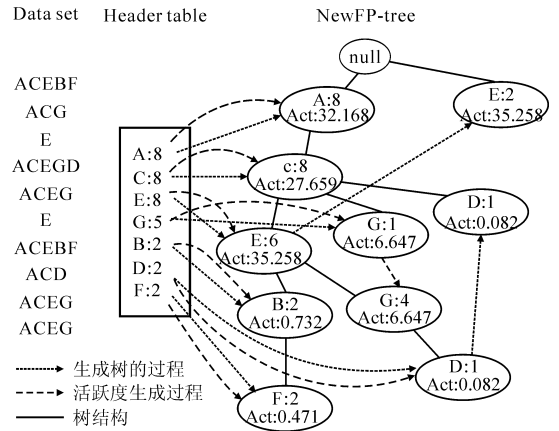


图2 IFP-tree 结构图

图2中,假设我们已经通过动态支持度建立了项头表,同时每个项目的兴趣值也包含于项头表中.那么虚线细箭头代表从项头表读出数据生成树的过程,黑色直线表示树的结构.虚线粗就是活跃度的生成过程,颜色浅的活跃度表示在此之前已经计算过该项目的活跃度.图中项目E、A和C活跃度明显靠前,可以认为这些项目更容易被大众所接受.

### 2.5 推荐系统实现

通过以上改进算法构建的 IFP-tree,为推荐系统提供了有效的推荐候选项.从 IFP-tree 中以用户最感兴趣的项目作为基点,寻找相关项目的最大频繁项,构成推荐序列,从相似度排序得到最终的推荐序列.

## 3 仿真与性能评价

在本节中,将本文提出的算法结合 EOS 区块链上的数据进行了实验验证.整个实验过程主要如下:采用爬虫技术在 EOS 区块链浏览器上爬取 762 个用户账号从 2018 年 08 月到 2018 年 12 月的总共

(242667)条浏览记录,其中项目数量为 1358 个;将这些数据放入本文的算法进行训练;用这些用户从 2019 年 01 月到 2019 年 02 月的浏览记录来进行验证,检验本文改进的算法的有效性.

### 3.1 性能评价指标

推荐算法中常用准确率(Precision)和召回率(Recall)值作为评价指标来验证算法的有效性. 准确率和召回率计算公式如下:

$$precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}$$

(7)

$$recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}$$

(8)

在推荐系统中,式(7)准确率表示推荐的项目真正为用户有过的行为数目占推荐的总项目的比例;式(8)召回率表示推荐的商品真正为用户有过的行为数目占用户实际有过行为项目的比例越大说明推荐效果越好.

$F_1$  分数同时兼顾准确率和召回率,更能体现推荐算法的有效性,计算如式(9):

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

(9)

### 3.2 最佳推荐数量的确定

本文从推荐序列中截取一定数量的项目推荐给用户. 推荐的数量过少,可能出现推荐信息不全面,推荐不准;推荐的越多,涵盖的信息越广,但会给用户造成选择困扰,甚至出现推荐饱和. 因此选择合适的推荐数量,有助于提高推荐效果.

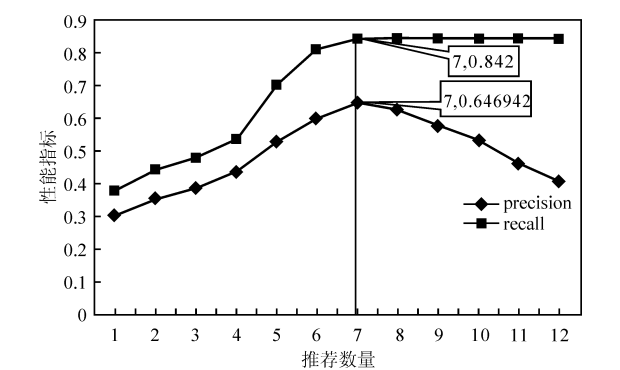


图 3 改变推荐数量对应性能指标

本实验通过改变推荐数量,观察准确率和召回率的变化,选取准确率取到峰值且召回率首次达到峰值对应的值作为最优的推荐数量. 如图 3 所示,可以发现当推荐数量等于 7 时,准确率最高,此时推荐效果最佳.

### 3.3 性能评估

确定好最佳推荐数量为 7,将我们的算法和传统 FP-growth 算法、矩阵分解算法和随机推荐进行性能对比. 通过实验分别计算出各自的准确率、召回率和  $F_1$  分数值,结果如图 4 到图 6 以及表 3.

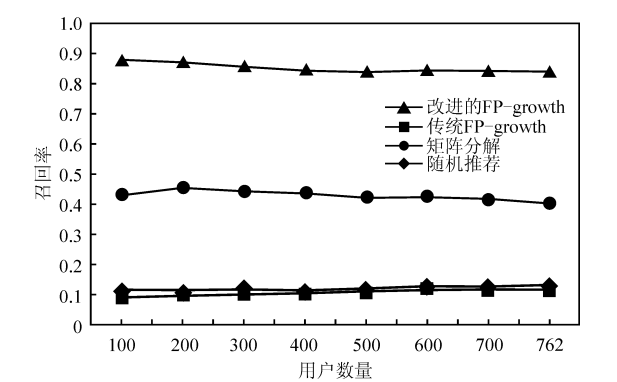


图 4 不同用户数量下推荐精度

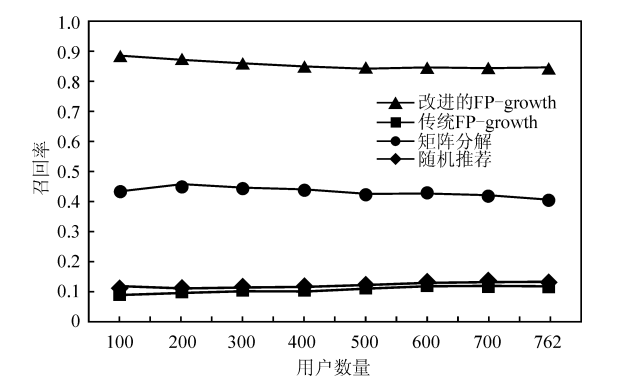


图 5 不同用户数量下召回率

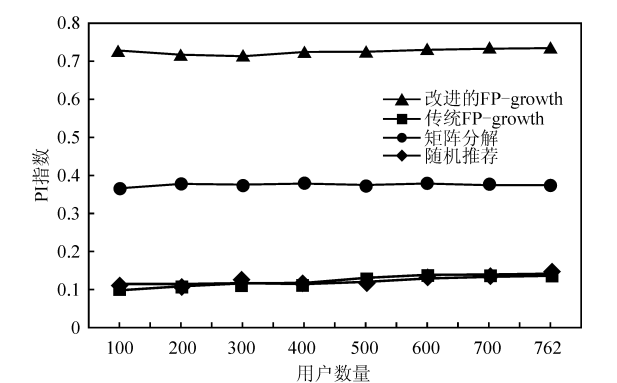


图 6 不同用户数量下 F1 指数

表 3 用户量为 762 时的几种算法性能评估

| 方法            | 准确率   | 召回率   | F1 指数 |
|---------------|-------|-------|-------|
| 改进的 FP-growth | 0.646 | 0.842 | 0.732 |
| 传统 FP-growth  | 0.158 | 0.118 | 0.135 |
| 矩阵分解          | 0.347 | 0.404 | 0.373 |
| 随机推荐          | 0.141 | 0.130 | 0.135 |

从图 4 到图 6 的总体变化曲线可以看出,对于改进的 FP-growth 算法,其算法性能相对于其他算法提升接近 30%,这主要归功于本文的改进不仅考虑到属于用户兴趣特征的项目,还加入遗忘函数修正用户兴趣在时间上兴趣迁移,使算法能够调整这些项目在 IFP-tree 中的影响权重,同时随着用户量的增加,信息提纯越高,推荐也越精准. 表 3 是几种算法性能对比的混淆矩阵,其中用户量为 762. 从表中可以直接看出,本文的算法性能优越,足以实现理想的推荐精度.

关于矩阵分解算法,其各项指标都在 0.35 左右,效果一般. 主要原因是该算法不能跟踪用户兴趣变化以及矩阵稀疏度的限制. 对于传统 FP-growth 和随机推荐,可以发现各项指标都在 0.1~0.2 之间,效果较差,因为传统 FP-growth 算法在建立项头表时,不能考虑每个用户的项目的特殊性;随机推荐更是没有规律,两种方法推荐效果都不理想.

3.4 非兴趣项目的过滤性能

在超过 1000 个项目中,标记其中 50 个典型的非兴趣项目. 改变推荐用户量,获得算法结果中的非兴趣项残留值. 实际结果如图 7 所示.

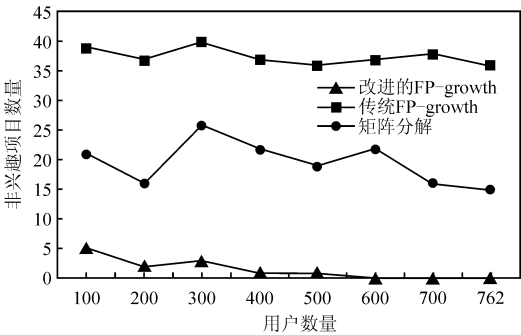


图 7 不同用户数量下非兴趣项目剩余

从图 7 中可以看出,三种算法曲线变化规律基本相同:随着用户数量的增加,非兴趣项目呈减少趋势. 同时本文提出的算法性能最好,主要是因为该算法能够将非兴趣项目、过时的数据过滤掉. 增加用户量,当前受欢迎的项目被凸显. 相反,传统 FP-growth 算法滤除标记的非兴趣项性能最差,因为只要项目达到支持度阈值,就会永远存留在 FP-tree 中. 矩阵分解算法的过滤性能一般,因为当两个用户兴趣相似时,非兴趣项目可能被推荐.

3.5 冷启动性能评估

对于没有用户额外信息推荐系统中的冷启动问题,本文也提出了一种解决方案. 如前文提到,根据 IFP-tree 上活跃度最高的项目作为推荐候选. 本文

用 87 个从 2019 年 1 月 1 日之后加入的用户的浏览记录,来验证本文提出的算法解决冷启动的能力. 调整推荐数量为 7,实验结果如图 8 所示.

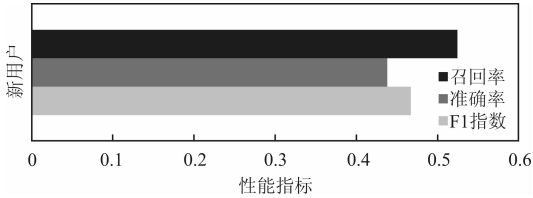


图 8 新用户冷启动性能

其中召回率为 0.525,准确率为 0.437,F1 指数为 0.467,这个结果体现出本文的算法在面对新加入的用户,推荐性能也比较良好. 由于用户信息的缺失,推荐系统常常无法把握用户的具体兴趣偏好,因此推荐指标不如对老用户的推荐指标.

3.6 结果比较及分析

实验表明,基于 FP-growth 改进的算法各项指标理想,该算法推荐的项目更为接近用户的行为. 同时,由于该算法推荐的项目都是用户没有接触过的,如果将用户当前感兴趣项目继续反馈给用户,那么性能指标可能还会有所提升.

4 结束语

本文根据用户的历史浏览记录研究用户兴趣推荐的问题. 该系统不仅可以当前用户的兴趣偏好提供相应的推荐,还可以考虑每个兴趣特征在时间上的变化,有助于捕捉当前最热门的项目. 同时也有效解决了本研究系统的冷启动问题. 后续研究可以考虑利用用户信息以及评分矩阵,结合协同过滤研究更精确的推荐.

参考文献:

[1] 许海玲, 吴潇, 李晓东, 等. 互联网推荐系统比较研究[J]. 软件学报, 2009, 20(2):350-362.

[2] SHARIFI Z, REZGHI M, NASIRI M. A new algorithm for solving data sparsity problem based-on Non negative matrix factorization in recommender systems [C]// International Econference on Computer & Knowledge Engineering. IEEE, 2014:56-61.

[3] REN Y, LI G, ZHANG J, et al. Lazy Collaborative Filtering for Data Sets With Missing Values[J]. IEEE Transactions on Cybernetics, 2013, 43(6):1822-1834.

[4] MINAEI-BIDGOLI B, ESMAEILI L, NASIRI M. Comparison of group recommendation techniques in social networks[C]// International Econference on Com-

- puter & Knowledge Engineering. IEEE, 2013: 236-241.
- [5] ESMAEILI L, NASIRI M, MINAEI-BIDGOLI B. Personalizing Group Recommendation to Social Network Users[C]// Web Information Systems and Mining - International Conference, WISM 2011, Taiyuan, China, September 2011:24-25.
- [6] YE H, ZHANG Q, BAI X. A new collaborative filtering algorithm based on modified matrix factorization [J]. 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, 2017:147-151.
- [7] YU H F, HSIEH C J, SI S, et al. Scalable Coordinate Descent Approaches to Parallel Matrix Factorization for Recommender Systems[C]// 2012 IEEE 12th International Conference on Data Mining. IEEE Computer Society, 2012:765-774.
- [8] ZHANG C, YANG M, LV J, et al. An Improved Hybrid Collaborative Filtering Algorithm Based on Tags and Time Factor[J]. Big Data Mining and Analytics, 2018, 1(2):48-56.
- [9] HAN Y, XIA K. Data Preprocessing Method Based on User Characteristic of Interests for Web Log Mining[C]// Fourth International Conference on Instrumentation & Measurement. IEEE, 2014:876-872.
- [10] GRAHNE G, ZHU J. Fast Algorithms for Frequent Itemset Mining Using FP-Trees[J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(10):1347-1362.
- [11] AHMED C F, TANBEER S K, JEONG B S, et al. Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases [J]. IEEE Transactions on Knowledge and Data Engineering, 2009, 21(12): 1708-1721.
- [12] YIN H, ZHOU X, CUI B, et al. Adapting to User Interest Drift for POI Recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2016: 1-1.
- [13] SEGATORI A, BECHINI A, DUCANGE P, et al. A Distributed Fuzzy Associative Classifier for Big Data [J]. IEEE Transactions on Cybernetics, 2017:1-14.
- [14] HAN J, PEI J, YIN Y. Mining Frequent Patterns Without Candidate Generation[C]// Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 2000:16-18.
- [15] HAN J, PEI J, YIN Y, et al. Mining Frequent Patterns without Candidate Generation; A Frequent-Pattern Tree Approach[J]. Data Mining and Knowledge Discovery, 2004, 8(1):53-87.
- [16] 贡佳炜, 单明辉, 陈君, 等. 声望模型中一种动态选择遗忘因子的方法[J]. 计算机工程与应用, 2009(17):23-25.
- [17] HAN J, PEI J, YIN Y. Mining Frequent Patterns Without Candidate Generation[C]// Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 2000:16-18.

### 作者简介:

彭浩男, (1994), 硕士研究生. 研究方向为机器学习与机器视觉检测.

佃松宜(通讯作者)男, (1972), 博士, 教授. 研究方向为先进智能控制. E-mail: scudiansy@mail.scu.edu.cn