软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2015,26(6):1395−1408 [doi: 10.13328/j.cnki.jos.004587]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种解决新项目冷启动问题的推荐算法*

于 洪, 李俊华

(计算智能重庆市重点实验室(重庆邮电大学),重庆 400065)

通讯作者: 于洪, E-mail: yuhong@cqupt.edu.cn, http://cs.cqupt.edu.cn//yuhong

摘 要: 推荐系统作为缓解信息过载问题的有效方法之一,在社交媒体中的作用日趋重要.但是,新项目冷启动和新用户冷启动问题是推荐技术面临的难题.为了解决新项目冷启动问题,提出了用户时间权重信息概念,该定义考虑到了用户评价时间与项目发布时间的时间间隔,根据用户时间权重值的大小,可以判断该用户是积极用户还是消极用户,以及用户对新项目的偏爱程度;利用三分图的形式来描述用户-项目-标签、用户-项目-属性之间的关系.在充分考虑用户、标签、项目属性、时间等信息基础上,获得个性化的预测评分值公式,提出了推荐算法.实验结果表明:所提出的方法能够实现满足不同用户、不同偏好的个性化推荐,在为用户推荐到合适项目的同时还能带来惊喜.比较实验说明,所提出的方法推荐准确度高,推荐新颖度高.交叉验证实验结果表明:该方法在解决推荐算法中的新项目冷启动问题上,无论是在推荐的准确度还是推荐项目的新颖度上都是有效的.

关键词: 推荐系统;协同过滤;冷启动;个性化;标签

中图法分类号: TP18

中文引用格式: 于洪,李俊华.一种解决新项目冷启动问题的推荐算法.软件学报,2015,26(6):1395−1408. http://www.jos.org.cn/1000-9825/4587.htm

英文引用格式: Yu H, Li JH. Algorithm to solve the cold-start problem in new item recommendations. Ruan Jian Xue Bao/ Journal of Software, 2015,26(6):1395−1408 (in Chinese). http://www.jos.org.cn/1000-9825/4587.htm

Algorithm to Solve the Cold-Start Problem in New Item Recommendations

YU Hong, LI Jun-Hua

(Chongqing Key Laboratory of Computational Intelligence (Chongqing University of Posts & Telecommunications), Chongqing 400065, China)

Abstract: As one of the effective methods to ease the information overload problem, recommender systems have become extremely popular in social media. However, recommender methods suffer from the cold-start problems in new item recommendations and new user recommendations. To combat the cold-start problems in new item recommendations, the concept of user time weights is proposed to characterize the time interval between the user evaluating time and item distributing time. According to the weights, it can determine whether the user is a positive user or a negative user, and the degree of the user's preference for new items. Tripartite graphs are used to picture relations among user-item-tag, and user-item-attribute. Combing information among users, tags, attributes of items and time weights, functions for predicting the rating are defined and a new personalized recommendation algorithm is constructed. Overall experimental results show that the proposed method not only brings satisfied personalized items but also pleasantly surprises different users with different preferences. Comparative experiments illustrate the proposed method is much higher in accuracy and novelty. Cross-validation experiments demonstrate that the new method is effective to solve the cold-start problem in new item recommendations.

Key words: recommender system; collaborative filtering; cold-start; personalization; tag

近年来,社交媒体越来越流行,在线用户的行为发生了巨大的变化.Web 用户不仅是信息的消费者,还是信息

* 基金项目: 国家自然科学基金(61379114, 61272060); 重庆市自然科学基金(cstc2011jjA40045)
收稿时间: 2012-11-14; 修改时间: 2013-06-26; 定稿时间: 2014-01-21

的生产者.为了解决信息过载问题,帮助用户快速准确地找到自己想要的东西,推荐系统应运而生,并被认为是缓解信息过载问题的最有效的方法之一[1].其中,协同过滤推荐算法是推荐系统中应用最广泛、最成功的推荐技术之一[2−4].但是,如果一个新的项目在评分矩阵中没有任何用户为它进行评价,或者是一个新用户在评分矩阵中没有对任何项目进行过评价,则无法使用协同过滤推荐算法实现推荐,这就是协同过滤推荐算法中经典的冷启动问题[5].对于电子商务推荐系统而言,总是不断地有新商品加入到系统中来,如果新商品能够及时地被推荐给合适的用户,可以提高新商品的销售量和推荐系统的新颖度,为商家赢得更多的利润同时,也能为用户带来意外惊喜,必须有效地解决新项目的冷启动问题.

因此,本文针对协同过滤推荐算法中存在的经典问题之一——新项目冷启动问题进行了研究.特别需要指出的是:本文所指的新项目是指完全新项目,即不存在任何一个用户过去或现在对该项目的评价或标注信息.虽然目前已存在很多解决冷启动问题的推荐算法[6−10],但是这些文献或多或少都会用到用户对该项目的评价信息,更多的是解决了推荐算法中的数据稀疏性问题,并没有真正做到对完全新项目的推荐.

由于新项目不存在任何一个用户对该项目的评价信息,因此,本文提出解决冷启动问题的个性化推荐算法并不仅仅使用用户对项目的评分信息,而是综合考虑项目本身的属性信息、标签对项目的标注信息以及用户对项目的评分时效性信息,全方位地考虑影响推荐效果的诸多因素;特别是,本文利用了用户评价时间与项目发布时间间隔的信息,在某一角度直接反映了用户对项目的兴趣与爱好,并且用户时间权重信息与每个用户评价项目的时间与项目发布的时间相关,是动态变化的.所以,将用户时间权重信息、项目属性信息和项目标签信息加入到推荐算法中,能够根据用户自身的不同兴趣与爱好实现对用户的个性化推荐,从而解决个性化推荐算法中的新项目冷启动问题.实验结果表明:本文提出的方法能够实现满足不同用户、不同偏好的个性化推荐;同时,在为用户推荐到合适项目的基础上还能带来惊喜.比较实验说明:本文提出的方法推荐准确度高、推荐新颖度高.交叉验证实验结果验证了本文提出的新方法在解决推荐算法中的新项目冷启动问题上无论是推荐的准确度还是推荐项目的新颖度上都是有效的.

本文第 1 节介绍相关工作.第 2 节给出相关概念.第 3 节提出用户时间权重的概念,采用三分图的形式描述用户-项目-标签、用户-项目-属性之间的关系,定义一系列新的预测评分值公式,从而提出多种新的个性化推荐算法,并进行分析比较.第 4 节给出在公共数据集 Movielens 上进行的一系列实验结果与分析,验证本文提出的个性化推荐算法不仅提高了推荐精度,而且有效地解决了推荐算法中的新项目冷启动问题.最后给出总结.

1 相关工作

伴随 Web 2.0 的发展以及各种类型社交媒体的流行,使得用户与互联网的交互成为可能,网络信息共享方式由单一的下载方式转为主动发布方式[11],如博客、维基和书签等主动发布信息的方式已经成为网络信息共享的方式.虽然用户的个性化参与丰富了网络信息的来源,加速信息扩散,但同时也使得信息过载问题备受关注.个性化推荐技术作为解决网络信息过载问题的重要工具,近年来成为学术界和互联网企业界的研究热点.

由于社交媒体中可以获得丰富多样的数据,包括标签、用户的社交关系等,促使社交媒体中的推荐模式不再单一,推荐的内容更加多样化,除了包括一般的资源,如电影、音乐等,还包括标签和人的推荐.而基于协同过滤的推荐技术,考虑用户对项目的评分信息,挖掘用户与项目之间评价关系进行预测推荐.很明显,仅仅依靠用户-项目二元关系已经不能满足当今用户的个性化需求,因此,我们需要考虑影响推荐效果的其他信息资源.

标签是 Web 2.0 时代在社会网络中的重要应用,体现了用户对资源的理解,既表达了信息资源的主要特征,又涵盖了用户与资源之间以及用户与用户之间的关系,兼具内容与关联的特征[12].Delicious 共享书签、Flickr 共享照片、CiteULike 共享学术文献等都应用了社会标签来描述和共享资源.将标签作为推荐技术的数据来源,充分利用自发标签直接反映用户兴趣和资源内容的特点,便有可能开发出同时具备内容过滤和协同过滤优越性的推荐技术,提高推荐系统的准确性和交互性.Zhang 等人[9,10,14]介绍了基于用户-项目-标签三元组的个性化推荐算法,将标签信息作为一个重要角色应用到推荐算法中,并以三分图的形式来描述用户、项目与标签三者之间的关系,最终实现个性化推荐.Jomsri 等人[13]提出了基于标签的研究论文推荐系统的架构,利用标签集来表

达用户的偏好,并应用此偏好为用户推荐适合的研究论文.实验结果表明:用户自定义的标签能用于表达每个个体用户的偏好,提高了推荐准确度.另外,用户兴趣对时间是敏感的,推荐系统应该随着时间的变化为用户提供不同的个性化推荐结果,目前在推荐过程中考虑时间因素的研究还较少.Koren[15]基于协同过滤的两种方法,在因子模型中建模用户偏差、对象偏差和用户兴趣的时间变化,目的是从数据中提取一些影响用户偏好的长期因素,在邻域模型中也建模了用户偏差、对象偏差的时间变化,还考虑用户打分的时间不同的情况,目的是发现基本的对象关联关系.通过这两种模型在 Netflix 的数据中进行实验,得到的结果都比不考虑时间影响的情况有了显著提高.Xiang 等人[16]分别对用户的长期和短期偏好进行建模,将用户某一时刻之前的选择作为长期偏好,集成长期和短期偏好形成推荐.实验结果表明,此方法取得了更高的准确率.Zheng 等人[17]研究了在社会化标注环境下,使用标签和时间信息来预测用户偏好的重要性,建立了基于这些信息的项目推荐模型,并在实验室中证实了将标签和时间信息集成到协同过滤推荐系统中可以提高推荐系统的准确度.

随着研究的深入以及推荐系统的应用领域广泛拓展,推荐系统也面临一系列挑战.而作为个性化推荐系统应用最为成功的技术之一的协同过滤技术,其最大的优点在于不需要分析项目的特征属性,对推荐对象没有特殊的要求,因而能够处理非结构化的复杂对象.这些优点使得协同过滤技术在理论研究和实际应用上都取得了很大的发展,但同时也存在着数据稀疏性、冷启动和隐私保护等问题[18,19],需要不断完善和解决.其中,冷启动问题又分为新项目推荐和新用户推荐.如果一个新项目没有人去评价它,则该项目肯定得不到推荐,推荐系统就失去了作用;同样,如果一个新用户从来没有对系统中的任何项目进行过评价,则系统无法获知其兴趣偏好,进而无法确定其近邻用户,也就无法对其进行推荐.

目前,研究学者提出了很多解决冷启动问题的方法.文献[6]针对现有服务选择中服务推荐技术的不足,提出一种基于偏好推荐的服务选择方法,解决推荐算法中的冷启动、推荐信息不准确等问题.但是文中没有利用服务本身的属性信息,只是单纯从用户间关系及用户偏好的角度来考虑.文献[7]针对冷启动问题,先构造出用户和资源的类别模型,构造出用户资源对来标记出用户感兴趣的资源.对于新用户,根据其一些重要的属性特征,把他分到对应的用户类别;对于新的资源,利用贝叶斯分类方法将其分到对应的资源类别模型.该文最后实验部分对于分类效果给出了数据分析,但是对于冷启动问题只是给出了合乎常识的推断,没有精确地实验数据做依据.文献[8]针对基于矩阵分解的协同过滤推荐算法中新用户和新项目的冷启动问题,通过运用基于 k 近邻的属性-特征映射的算法得到新用户和新项目的特征向量,解决了该类协同过滤算法面临的冷启动问题.但是文中提出的方法比较局限,只适用于基于矩阵分解的协同过滤推荐算法.文献[9,10]介绍了一种基于用户-项目-标签三元组的个性化推荐算法,将标签信息作为一个重要角色应用到推荐算法中,并以三分图的形式来描述用户、项目与标签三者之间的关系,最终实现推荐,以解决推荐算法的冷启动问题.但是文献中并没有彻底做到解决新项目冷启动的问题,因为真正的新项目是不存在用户对该项目的评分信息和标签信息的,所以文献中的推荐算法对于新项目来说,计算出来的预测评分值为 0.也就是说,使用文献[9,10]中的算法新项目永远不会得到被推荐的机会.

2 有关基本概念

2.1 传统协同过滤推荐算法

首先,传统的协同过滤推荐算法中,用户对项目的评分信息使用 $m\times n$ 阶的用户-项目评分矩阵表示.一般说来,使用$\{u_1,u_2,\ldots,u_m\}$表示 m 个用户的集合,$\{i_1,i_2,\ldots,i_n\}$表示 n 个项目的集合,r_{ij} 表示用户对项目的评分值.表 1 给出了用户-项目评分矩阵例子.然后,就可以计算用户或项目间的相似度.常见的相似度计算方法有 3 种:余弦相似性、皮尔森相关相似性和修正的余弦相似性[20].其中,皮尔森相关相似性如下:

$$sim(u,v)=\frac{\sum_{i\in P_{uv}}(r_{ui}-\overline{r}_u)(r_{vi}-\overline{r}_v)}{\sqrt{\sum_{i\in P_{uv}}(r_{ui}-\overline{r}_u)^2}\sqrt{\sum_{i\in P_{uv}}(r_{vi}-\overline{r}_v)^2}} \tag{1}$$

公式(1)中,$\overline{r}_u$ 和 $\overline{r}_v$ 分别表示用户 u 和 v 的评分平均值,P_{uv} 表示用户 u 和 v 共同评分的项目集合.

Table 1 Users-Items rate matrix

表 1 用户-项目评分矩阵

r_{ij}	i_1	i_2	i_3	i_4
u_1	2	0	4	0
u_2	0	3	1	0
u_3	5	0	2	0

最后,使用预测评分公式(2)计算得到目标用户对项目的预测评分值,实现推荐:

$$P_{ui}=\overline{r}_u+\frac{\sum_{v\in NBS_u} sim(u,v)\times(r_{vi}-\overline{r}_v)}{\sum_{v\in NBS_u} sim(u,v)} \tag{2}$$

公式(2)中,P_{ui}表示用户 u 对项目 i 的预测评分值,$\overline{r}_u$ 和 $\overline{r}_v$ 分别表示用户 u 和 v 的评分平均值,r_{vi}表示用户 v 对项目 i 的评分,NBS_u表示用户 u 的近邻集合.

2.2 项目属性、项目标签与用户时间信息

项目属性信息使用 $p\times n$ 阶的属性-项目矩阵表示,其中,$\{a_1,a_2,\ldots,a_p\}$表示 p 个属性的集合,$\{i_1,i_2,\ldots,i_n\}$表示 n 个项目的集合,h_{ij}的值表示项目是否具备该属性.表 2 给出了属性-项目矩阵例子.

标签对项目的标注信息使用 $q\times n$ 阶的标签-项目标注矩阵表示,其中,$\{t_1,t_2,\ldots,t_q\}$表示 q 个标签的集合,$\{i_1,i_2,\ldots,i_n\}$表示 n 个项目的集合,b_{ij}表示项目被标签标注的次数.表 3 给出了标签-项目标注矩阵例子.

Table 2 Attributes-Items matrix

表 2 属性-项目矩阵

h_{ij}	i_1	i_2	i_3	i_4
a_1	1	0	1	0
a_2	1	1	1	0
a_3	1	0	1	1
a_4	0	1	0	1
a_5	0	1	0	1

Table 3 Tags-Items dimension matrix

表 3 标签-项目标注矩阵

b_{ij}	i_1	i_2	i_3	i_4
t_1	2	3	0	0
t_2	0	3	1	0
t_3	1	0	0	0
t_4	0	2	1	0
t_5	0	0	2	0

3 多种个性化推荐算法

3.1 用户时间权重

现实生活中,有些用户喜欢去关注并评价新事物,我们称这类用户为积极用户;有的用户可能比较喜欢去关注已经被很多用户评价过的事物,我们称这类用户为消极用户.对于积极用户来说,他们评价项目的时间与项目发布的时间间隔一般会很短;而对于消极用户来说,他们评价项目的时间与项目发布的时间间隔会相对较长.一般说来,在推荐新项目的过程中,积极用户和消极用户都会存在,我们会优先选择将新项目推荐给积极用户.鉴于此,本文将时间信息加入到推荐算法中,解决推荐算法中的冷启动问题.

为了描述时间信息对推荐算法的影响,首先定义用户时间权重如下:

定义 1(用户时间权重). 用户评价项目的总数与用户评价过的所有项目的时间与项目发布时间间隔的和的比值.

这里,定义 sum 表示用户 u 评价项目的总数;$time_{ui}$表示用户 u 评价项目 i 的时间;$date_i$表示项目 i 的发布时间.则用户 u 的时间权重 w_u 计算公式为

$$w_u=\frac{sum}{\sum_{i=1}^{sum}(time_{ui}-date_i)} \tag{3}$$

公式(3)中,$(time_{ui}-date_i)$的值越小,w_u 越大,表示用户评价项目的时间与项目发布的时间距离越近,说明该用户越积极,喜欢去关注新事物;反之,因式$(time_{ui}-date_i)$的值越大,w_u 越小,说明该用户比较消极,喜欢去关注已经被很多用户关注或评价过的项目.公式(3)反映了用户偏爱评价新发布项目的平均程度.

接下来,我们以用户 u_1,u_2,u_3 和项目 i_1,i_2,i_3,i_4 为例来描述用户时间权重信息.其中,time 表示用户评价项目的时间,date 表示项目发布的时间,时间单位为年,用户评价项目的时间信息见表 4.

Table 4 User appraisal schedule

表 4 用户评价时间表

Item	u_1		u_2		u_3	
	Time	Date	Time	Date	Time	Date
i_1	2006	1991	2008	1991	2002	1991
i_2	2006	2003	2008	2003	2009	2003
i_3	2005	2005	2009	2005	2006	2000
i_4	0	2002	0	2002	0	2002

以表 4 为例,采用公式(3)计算得到每个用户的时间权重值 w_u,即:

$$w_{u_1}=\frac{sum}{\sum_{i=1}^{sum}(time_{u_1i}-date_i)}=\frac{3}{15+3+0}=0.17,$$

$$w_{u_2}=\frac{sum}{\sum_{i=1}^{sum}(time_{u_2i}-date_i)}=\frac{3}{17+5+4}=0.12,$$

$$w_{u_3}=\frac{sum}{\sum_{i=1}^{sum}(time_{u_3i}-date_i)}=\frac{3}{11+6+6}=0.13.$$

根据上面计算的结果 $w_{u_1}>w_{u_3}>w_{u_2}$,我们可知:用户 u_1 相比用户 u_2 和 u_3 来说更偏爱评价新发布的项目,而用户 u_2 和 u_3 则比较偏爱评价或关注那些已经被其他用户关注或已经发布很久的项目.

3.2 用户预测评分值

这里,我们假定以用户 u 为研究对象.分析用户-项目评分矩阵的信息,可以得到用户 u 评价项目的个数和项目 j 被用户评价的次数.令 r_{uj} 表示用户 u 是否评价项目 j,1 表示评价过,0 表示没有评价过;IC_k 表示用户 k 评价的项目个数,UC_j 表示共同评价项目 j 的用户个数.那么,根据用户对项目的评分信息以及用户评价项目的个数和项目被用户评价的次数信息,可以定义基于项目评分信息的预测评分值,从而挖掘他们之间的关系.

定义 2(基于项目评分信息的预测评分值). 基于项目评分信息的预测评分值计算公式 $f_u^j(u)$ 如下:

$$f_u^j(u)=\sum_{k=1}^{m}\frac{1}{IC_k}\sum_{j=1}^{n}\frac{r_{uj}\cdot r_{kj}}{UC_j} \tag{4}$$

公式(4)中, $f_u^j(u)$ 表示用户 u 对项目 j 的预测评分值.如果用户 u 对项目 j 做了评价,则 r_{uj}=1;反之,r_{uj}=0.如果用户 k 对项目 j 做了评价,则 r_{kj}=1;反之,r_{kj}=0.分析项目-属性矩阵的信息,可以得到项目 j 具有的属性个数和共同具有属性 l 的项目的个数.h_{lj} 表示项目 j 是否具有属性 l,1 表示具有,0 表示不具有;AC_j 表示项目 j 具有的属性个数;IAC_l 表示共同具有属性 l 的项目个数.那么,根据用户对项目的评分信息、项目具备的属性信息、项目具有属性的个数以及共同具有属性的项目的个数信息,可以定义基于项目属性信息的预测评分值,从而挖掘他们之间潜在的关系.

定义 3(基于项目属性信息的预测评分值). 基于项目属性信息的预测评分值计算公式 $f_a^j(u)$ 如下:

$$f_a^j(u)=\sum_{l=1}^{p}\frac{1}{IAC_l}\sum_{j=1}^{n}\frac{r_{uj}\cdot h_{lj}}{AC_j} \tag{5}$$

公式(5)中, $f_a^j(u)$ 表示基于属性的用户 u 对项目 j 的预测评分值.如果用户 u 对项目 j 做了评价,则 r_{uj}=1;反之,r_{uj}=0.如果项目 j 具有属性 l,则 h_{lj}=1;反之,h_{lj}=0.

分析项目-标签标注矩阵的信息,可以得到项目被标签标注的次数和标签 g 标注项目的个数.令 b_{gi} 表示标签 g 是否标注项目 j,1 表示标注过,0 表示未标注;TC_j 表示项目 j 被标注的标签的个数;ITC_g 表示标签 g 标注的项目

个数.那么,根据用户对项目的评分信息、标签对项目的标注信息、项目被标签标注的次数以及标签标注项目的个数信息,可以定义基于项目标签信息的预测评分值,从而挖掘他们之间的潜在的关系.

定义 4(基于项目标签信息的预测评分值). 基于项目标签信息的预测评分值计算公式 $f_t^j(u)$ 如下:

$$f_t^j(u)=\sum_{g=1}^{q}\frac{1}{ITC_g}\sum_{j=1}^{n}\frac{r_{uj}\cdot b_{gj}}{TC_j} \tag{6}$$

公式(6)中, $f_t^j(u)$ 表示基于标签的用户 u 对项目 j 的预测评分值.如果用户 u 对项目 j 做了评价,则 r_{uj}=1;反之,r_{uj}=0.如果标签 g 标注了项目 j,则 b_{gi}=1;反之,b_{gi}=0.

为了更好地理解公式(4)、公式(5)和公式(6),以表 1、表 2 和表 3 中的用户、项目、项目属性和标签信息为例,使用两个三分图的形式来表示,如图 1 和图 2 所示.在图 1 和图 2 中,u 表示用户,i 表示项目,a 表示项目的属性,t 表示标签.如果用户 u 评价了项目 i,则用户 u 与项目 i 之间使用一条线进行连接,边上的值为用户 u 对项目 i 的评分值;项目 i 与标签 t 和项目 i 与属性 a 之间的关系是类似的.其中,i_4 为新项目,即,该项目不存在用户评价信息以及标签的标注信息.

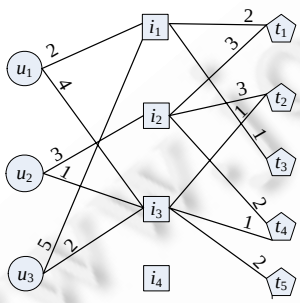


Fig.1 Users-Items-Tags information

图 1 用户-项目-标签信息

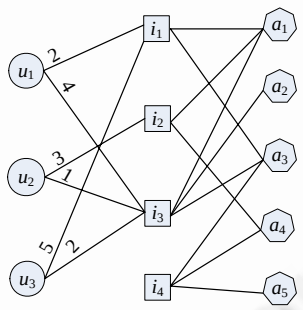


Fig.2 Users-Items-Attributes information

图 2 用户-项目-属性信息

这里,我们假定以用户 u_1 为研究对象.首先使用公式(4)计算得到所有项目基于用户 u_1 的评分信息的预测评分值,如下所示:

$$f_{u_1}=r_{11}\times r_{11}\times\frac{1}{UC_1}+r_{12}\times r_{12}\times\frac{1}{UC_2}+r_{13}\times r_{13}\times\frac{1}{UC_3}+r_{14}\times r_{14}\times\frac{1}{UC_4}=1\times1\times\frac{1}{2}+0\times0\times1+1\times1\times\frac{1}{3}+0\times0\times0=\frac{5}{6},$$

$$f_{u_2}=r_{11}\times r_{21}\times\frac{1}{UC_1}+r_{12}\times r_{22}\times\frac{1}{UC_2}+r_{13}\times r_{23}\times\frac{1}{UC_3}+r_{14}\times r_{24}\times\frac{1}{UC_4}=1\times0\times\frac{1}{2}+0\times1\times1+1\times1\times\frac{1}{3}+0\times0\times0=\frac{1}{3},$$

$$f_{u_3}=r_{11}\times r_{31}\times\frac{1}{UC_1}+r_{12}\times r_{32}\times\frac{1}{UC_2}+r_{13}\times r_{33}\times\frac{1}{UC_3}+r_{14}\times r_{34}\times\frac{1}{UC_4}=1\times1\times\frac{1}{2}+0\times0\times1+1\times1\times\frac{1}{3}+0\times0\times0=\frac{5}{6},$$

$$f_{u_1}^{i_1}(u_1)=f_{u_1}\times\frac{1}{2}+f_{u_3}\times\frac{1}{2}=\frac{5}{6}\times\frac{1}{2}+\frac{5}{6}\times\frac{1}{2}=\frac{5}{6},$$

$$f_{u_1}^{i_3}(u_1)=f_{u_1}\times\frac{1}{2}+f_{u_2}\times\frac{1}{2}+f_{u_3}\times\frac{1}{2}=\frac{5}{6}\times\frac{1}{2}+\frac{1}{3}\times\frac{1}{2}+\frac{5}{6}\times\frac{1}{2}=1,$$

$$f_{u_1}^{i_4}(u_1)=0.$$

其中, $f_{u_1}^{i_1}(u_1)$ 表示通过分析用户 u_1 对项目 i_1 评分信息以及用户评价项目的个数和项目被用户评价的次数信息,获得的用户 u_1 对项目 i_1 的预测评分值; f_{u_1} 表示计算用户 u_1 对项目 i_1 评分值过程中的一个中间值.

接下来的公式计算过程中,我们以项目 i_2 为研究对象,同时在以用户 u_1 为研究对象的基础上进行.

然后,使用公式(5)计算得到所有项目基于用户 u_1 的项目属性信息的预测评分值,如下所示:

$$f_a^{i_2}(u_1)=f_{a_1}\times\frac{1}{3}+f_{a_4}\times\frac{1}{2}=\frac{5}{6}\times\frac{1}{3}+0\times\frac{1}{2}=\frac{5}{18}.$$

最后,使用公式(6)计算得到所有项目基于用户 u_1 的项目标签信息的预测评分值,如下所示:

$$f_t^{i_2}(u_1)=f_{t_1}\times\frac{1}{2}+f_{t_2}\times\frac{1}{2}+f_{t_4}\times\frac{1}{2}=\frac{1}{2}\times\frac{1}{2}+\frac{1}{3}\times\frac{1}{2}+\frac{1}{3}\times\frac{1}{2}=\frac{7}{12}.$$

上面的计算过程,展示了以用户 u_1 和项目 i_2 为研究对象的前提下,如何使用本文定义的 3 种典型情况下的预测评分值公式,获得用户 u_1 对项目 i_2 的使用不同信息后的预测评分值.

但是,通过观察图 1 和图 2 可知:i_4 为新项目,即,项目 i_4 不存在任何用户评分信息和标签标注信息,只有自身的属性信息.因此,我们需要使用公式(5)计算得到项目 i_4 基于用户的项目属性信息的预测评分值,如下所示:

$$f_a^{i_4}(u_1)=f_{a_3}\times\frac{1}{3}+f_{a_4}\times\frac{1}{2}+f_{a_5}\times1=\frac{5}{6}\times\frac{1}{3}+0\times\frac{1}{2}+0\times1=\frac{5}{18}.$$

3.3 多种个性化推荐算法

鉴于前文的分析,本节将结合用户、标签、项目属性提出多种新的个性化推荐算法;并且通过实验发现,本文提出的多种个性化推荐算法提高了推荐系统的推荐准确度.大多数推荐算法都难以解决经典的难题——新项目的冷启动问题,但是我们分析发现:结合用户的评分时效性信息,可以有效地解决推荐系统的冷启动问题,同时又保证了推荐准确度.

为了更好地观察用户评分信息、项目属性信息、项目标签信息以及用户评分时效性信息在推荐算法中所产生的影响,我们定义权重α,β和γ,线性组合可以得到各种情况,其中,$\alpha+\beta+\gamma=1$.在此,我们给出 3 种典型情况:

(1) 结合用户、标签与项目属性的个性化推荐算法,其中,预测评分值计算公式如下:

$$Pre_{uj}=\alpha\times f_u^j(u)+\beta\times f_t^j(u)+\gamma\times f_a^j(u) \tag{7}$$

在此,为了观察用户、标签与项目属性信息在推荐算法中的影响,本文定义了 3 种不同的用户预测评分值计算方法.当$\alpha\neq0,\beta\neq0,\gamma\neq0$ 时,综合考虑用户、标签与项目属性信息进行用户预测评分值计算,即为考虑用户、标签与项目属性的个性化推荐算法(consider user,tag and item-attribute personalized recommendation algorithm,简称 CUTA);当$\alpha\neq0,\beta=0,\gamma\neq0$,时,只考虑用户与项目属性信息进行用户预测评分值计算,即为用户与项目属性信息的个性化推荐算法(consider user and item-attribute personalized recommendation algorithm,简称 CUA);当$\alpha\neq0,\beta\neq0,\gamma=0$ 时,只考虑用户与标签信息进行用户预测评分值计算,即为考虑用户与标签信息的个性化推荐算法(consider user and tag personalized recommendation algorithm,简称 CUT).

(2) 结合用户评分值、标签标注次数与项目属性信息的个性化推荐算法,预测评分值计算公式如下:

$$Pre=\alpha\times f(u)\times r+\beta\times f(u)\times b+\gamma\times f(u) \tag{8}$$

这里,r_{uj} 表示用户 u 对项目 j 的评分值,b_{tj} 表示标签 t 标注项目 j 的次数.在此,为了观察用户、标签与项目属性信息在推荐算法中的影响,本文定义了 3 种不同的用户预测评分值计算方法.当$\alpha\neq0,\beta\neq0,\gamma\neq0$ 时,综合考虑用户评分值、标签标注次数与项目属性信息进行用户预测评分值计算,即为考虑用户评分值、标签标注次数与项目属性信息的个性化推荐算法(consider user-rating、tag-number and item-attribute personalized recommendation algorithm,简称 CURTNA);当$\alpha\neq0,\beta=0,\gamma\neq0$ 时,只考虑用户评分值与项目属性信息进行用户预测评分值计算,即为考虑用户评分值与项目属性信息的个性化推荐算法(consider user-rating and item-attribute personalized recommendation algorithm,简称 CURA);当$\alpha\neq0,\beta\neq0,\gamma=0$ 时,只考虑用户评分值与标签标注次数信息进行用户预测评分值计算,即考虑用户评分值与标签标注次数信息的个性化推荐算法(consider user-rating and tag-number personalized recommendation algorithm,简称 CURTN).

(3) 结合用户、标签、项目属性与用户时间权重信息的个性化推荐算法,即,考虑用户、标签、项目属性与用户时间权重信息的推荐算法(consider user,tag,item-attribute and time personalized recommendation algorithm,简称 CUTATime),其中,预测评分值计算公式如下:

$$Pre_{uj}=\begin{cases}\alpha\times f_u^j(u)+\beta\times f_t^j(u)+\gamma\times f_a^j(u), & UC_j\neq0,\ \alpha\neq0,\beta\neq0,\gamma\neq0 \quad (9)\\ w_u+f_a^j(u), & UC_j=0 \quad (10)\end{cases}$$

其中,UC_j 表示共同评价项目 j 的用户个数,w_u 为用户 u 的时间权重.如果项目 j 为新项目,则 $UC_j=0$;反之,$UC_j\neq0$.

第 3 种个性化推荐算法在前面两种个性化推荐算法提高推荐准确度的基础上,结合了项目属性和用户时

间权重信息.一方面,不管是新项目还是已经被评价过的项目,它都会具备自身的属性信息;另一方面,用户时间权重信息是指用户评分时间与项目发布时间的时间间隔,与项目是否为新项目也没有关系,但用户时间权重信息又可以从另外一个角度直接反映用户对项目的偏好程度.因此,本文提出的解决冷启动问题的 CUTATime 个性化推荐算法是合乎情理的,相比前面两种个性化推荐算法,对新项目的推荐效果也应该会更好,后文的实验结果也验证了这个结论.

3.4 考虑时间信息的个性化推荐算法描述

鉴于本文提出了多种个性化推荐算法,这里,我们以考虑用户、标签、项目属性与用户时间权重信息的推荐算法(CUTATime)为例来描述个性化推荐算法的实现过程:

- 首先,根据用户-项目评分信息,考虑用户评价项目的个数和项目被用户评价的次数,获得基于项目评分信息的预测评分值;
- 其次,根据标签-项目项目信息,考虑标签标注项目的个数和项目被标签标注的次数,获得基于项目标签信息的预测评分值;
- 然后,根据项目-属性信息,考虑项目具备属性的个数和共同具备某属性的项目的个数,获得基于项目属性信息的预测评分值;
- 最后,通过分析用户评价项目的时间信息,获得用户时间权重信息,结合基于项目评分信息的预测评分值,基于项目标签信息的预测评分值和基于项目属性信息的预测评分值,实现推荐.

考虑用户、标签、项目属性与用户时间权重信息的个性化推荐算法(CUTATime)描述如下:

输入:目标用户 u,目标项目 j,用户-项目评分矩阵 $R_{m\times n}$,属性-项目矩阵 $R_{p\times n}$,标签-项目矩阵 $R_{q\times n}$,用户评价项目的时间 $time_{uj}$,项目发布的时间 $date_j$;

输出:目标用户 u 对项目 j 的预测评分值 Pre_{uj}.

Step 1. 根据用户-项目评分矩阵 $R_{m\times n}$,采用公式(4)计算基于项目评分信息的预测评分值 $f_u^j(u)$;

Step 2. 根据属性-项目矩阵 $R_{p\times n}$,采用公式(5)计算基于项目属性信息的预测评分值 $f_a^j(u)$;

Step 3. 根据标签-项目矩阵 $R_{q\times n}$,采用公式(6)计算基于项目标签信息的预测评分值 $f_t^j(u)$;

Step 4. 根据用户评价项目的时间 $time_{uj}$ 和项目发布的时间 $date_j$,采用公式(3)计算用户 u 的时间权重 w_u;

Step 5. 结合基于项目评分信息的预测评分值 $f_u^j(u)$、基于项目属性信息的预测评分值 $f_a^j(u)$、基于项目标签信息的预测评分值 $f_t^j(u)$ 和用户 u 的时间权重 w_u,采用公式(9)和公式(10)计算目标用户 u 对项目 j 的预测评分值 Pre_{uj}.

假设系统中有 m 个用户、n 个项目、p 个项目属性和 q 个标签,算法 CUTATime 中步骤 1 到步骤 3 的时间复杂度分别为 $O(mn)$,$O(pn)$和 $O(qn)$;步骤 4 需要访问 n 个项目的信息一遍;步骤 5 的开销很小.目标用户对项目的预测评分值 Pre_{uj} 计算的时间复杂度为 $O(mn)+O(pn)+O(qn)+O(n)$.最坏情况下,仍然是一个 2 次幂的多项式时间复杂度函数.

随着系统中用户数、项目数以及用户的评分信息和标签数的增加,需要存储的矩阵增大,推荐算法效率会降低.要解决这个问题,我们拟在将来的工作中从以下两方面考虑:提出时间复杂度在 2 阶以下的算法;结合其他信息,使得 R_{mn} 等矩阵在运算时的实际存储空间控制在一个合适的范围内,而不是在整个数据集的 $m\times n$ 的空间上进行,可以考虑先进行用户或者项目的聚类分析方面的预处理,而这些运算都可以离线进行,不影响在线推荐运算时间.

4 实验结果与分析

4.1 数据集预处理与度量标准

实验采用的数据集是 2011 年第 2 届国际推荐系统研讨会上公布的 Movielens 数据集[21],Movielens 数据集

由美国 Minnesota 大学的 GroupLens 研究小组创建并维护,该数据集包含了用户对电影的评分信息、电影的属性信息以及电影的标签信息.

由于原数据集包含较多的噪音数据,无法在实验中直接使用,我们首先选取数据集中用户评分时间与电影发布时间间隔大于等于 0 的数据记录,然后选取至少被 20 个用户共同评价过的电影以及至少评价过 20 个电影的用户,经过处理的数据集中有 1 992 个用户、3 310 部电影,最后在数据集中选择与 3 310 部电影对应的标签信息和属性信息.经过预处理后的数据集划分为训练集和测试集,其中,训练集占 50%,测试集占 50%.

本文采用推荐准确度和新颖度作为度量标准,验证个性化推荐算法的有效性.推荐准确度指取前 N 个(Top-N)推荐给目标用户,如果 Top-N 推荐列表中某个被推荐项目出现在了某目标用户的测试集(test)中,则表明生成了一个正确推荐[22],计算公式为 $precision=Hit/N$.这里,$precision$ 表示准确度,Hit 表示 Top-N 推荐列表中正确推荐的项目个数.

定义 5(新颖度). 推荐给目标用户的 Top-N 推荐列表中包含的新项目个数与推荐列表中项目个数的比值,计算公式为

$$novelty=Num/N.$$

上式中,$novelty$ 表示新颖度,Num 表示 Top-N 推荐列表中包含的新项目个数.

4.2 实验结果与分析

为了验证本文提出方法的有效性,我们进行了 4 类实验.首先,针对本文第 3.3 节提出的多种个性化推荐算法的推荐准确度进行了对比实验;然后,针对多种个性化推荐算法解决完全新项目的冷启动问题的有效性进行了对比实验;随后,就我们提出的解决冷启动问题算法的性能进行了对比实验;最后,采用交叉验证实验方法验证本文提出的个性化推荐算法解决冷启动问题的有效性.

文献[10]中提出的用于冷启动问题的个性化推荐算法是基于用户、标签信息的,该文采用了三分图的形式来描述用户、项目与标签三者之间的关系.考虑到本文的主要工作目的也是提出冷启动问题的解决方法,因此,后文实验中采用了文献[10]中的算法作为对比算法.为了描述方便,我们称文献[10]中的算法为 CUT,表示考虑用户、标签信息的个性化推荐算法.

实验 1. 不同推荐算法的推荐准确度对比.

首先,针对本文提出的多种个性化推荐算法与文献[10]中的推荐算法进行了推荐准确度的对比.在此,我们在个性化推荐算法以及文献[10]中的推荐算法中的参数 α,β 和 γ 的不同取值组合情况下,针对近邻个数为 5,10,15,20,30 的情况分别进行了多组实验.限于篇幅,表 5 只给出了 8 组参数的取值示例.

Table 5 Parameters

表 5 部分参数取值

组号	参数取值	
	α,β,γ	α,γ或者α,β
1	0.1,0.4,0.5	0.1,0.9
2	0.1,0.8,0.1	0.2,0.8
3	0.5,0.2,0.3	1,0
4	0.5,0.4,0.1	0.6,0.4
5	0.6,0.3,0.1	0.1,0.9
6	0.7,0.2,0.1	0.2,0.8
7	0.8,0.1,0.1	1,0
8	0.8,0.1,0.1	0.6,0.4

下面先以 CUTA,CURTNA 算法为例讨论.我们针对参数 α,β 和 γ 的 36 种组合情况进行了实验.观察实验结果发现:当 α 取值比较小时,推荐准确度比较大;但是 β 和 γ 的取值组合对推荐准确度并无明显影响.比如,α=0.1,β=0.8,γ=0.1 与 α=0.1,β=0.1,γ=0.8 这两组实验结果在近邻个数为 5 时的推荐准确度分别为 0.743,0.745.实验中我们还发现:在相同参数取值情况下,CUTA 算法在推荐准确度上优于 CURTNA 算法.

其他几种推荐算法只涉及到两个参数.有意思的是,本文分析实验结果发现了与前述实验结果类似的现象:

当α取值比较小时,推荐准确度比较大.实验结果说明:在不考虑新项目的情况下,用户信息权值取值比较小时,推荐准确度相对较大,说明在推荐准确度的影响因素中用户信息所占权重比较小;其他两个参数,在不同组合情况下的推荐准确度变化不大,说明本文提出的算法对参数依赖性并不强.

作为示例,图 3 给出了在第 1 组、第 2 组、第 4 组、第 7 组这 4 组参数时,6 种推荐算法在不同近邻个数情况下的实验结果.为了对比公平,这几种参数取值情况包括了每个算法的最好情况.比如,CUTA 算法最好的一组实验结果对应的是第 1 组参数.图 3 中的小图给出了 CUTA 算法在这 4 组参数情况下的实验结果.

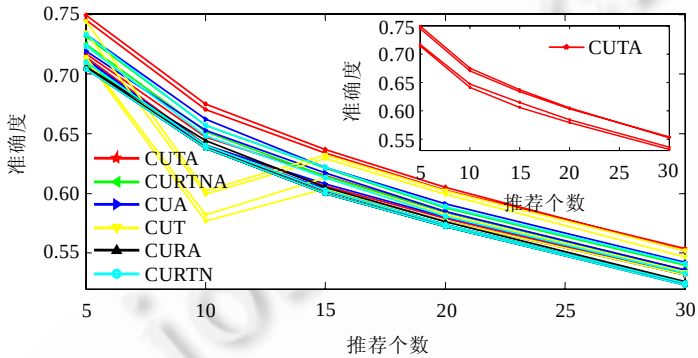


Fig.3 Precisions for different recommendation algorithms

图 3 不同推荐算法的准确度对比

观察图 3 可知,本文提出的考虑用户、标签与项目属性信息的个性化推荐算法的推荐准确度优于文献[10]中的 CUT 算法,其推荐准确度最高.因此,在接下来的实验中,我们采用 CUTA 推荐算法解决推荐系统的新项目冷启动问题.

实验 2. 解决冷启动问题:不同推荐算法的推荐准确度和新颖度对比.

由于新项目冷启动问题是用户未对项目进行评价,数据集中不存在用户评分信息以及标签信息,因此,我们需要通过考虑项目属性信息和用户评分时效性信息实现用户个性化推荐,提高新项目被推荐的机会.

首先,在实验 1 中进行实验的测试集中随机抽取 160 个项目作为新项目,训练集中对应的 160 个项目的评分信息及标签信息设为 0.然后,使用处理后的训练集和测试集进行实验,并与文献[10]中的推荐算法进行对比,这里,参数的α,β和γ也是如实验 1 按各种组合情况取值进行实验,比如,CUTA 和 CUTATime 算法也分别进行了 36 组实验.图 4 和图 5 分别记录了在不同近邻个数情况下,3 个算法在第 3 组、第 5 组、第 6 组、第 8 组参数取值情况时的准确度和新颖度.为了对比公平性,这几种参数取值情况同样包括了每个算法的最好情况.

观察实验结果发现:当数据集中新项目较多时,用户信息α的权重就显得比较重要了.相比于实验 1 来说,此时用户信息权重$\alpha\geqslant$0.5 时,推荐准确度会比较好些;β和γ的取值组合对推荐准确度并无明显影响.这个实验结果与我们在解决冷启动问题时采取的策略是一致的,即,通过考察用户对项目的关注程度来帮助实现.

分析实验结果可知:一方面,本文提出的 CUTA 推荐算法的推荐准确度略高于文献[10]中的 CUT 推荐算法,虽然 CUTATime 推荐算法仅在近邻取 5 时推荐准确度略高于文献[10]中的 CUT 推荐算法,但是当近邻取其他值时,与文献[10]中的 CUT 推荐算法的推荐准确度差距的范围是可以接受的;另一方面,本文提出的 CUTA 推荐算法和 CUTATime 推荐算法的推荐新颖度远远高于文献[10]中的 CUT 推荐算法,特别是 CUTATime 推荐算法的推荐新颖度值最高;尤其是,文献[10]中的算法并不能真正解决新项目的推荐问题,所以其新颖度是 0.综合考虑,本文提出的 CUTATime 推荐算法在处理新项目冷启动问题上推荐效果最佳.

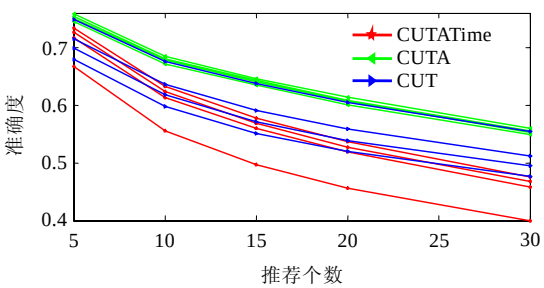


Fig.4 Precisions of algorithms solving cold-start problem

图 4 解决冷启动问题时算法的准确度对比

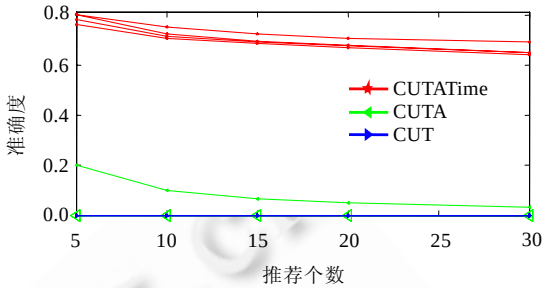


Fig.5 Novelties of algorithms solving cold-start problem

图 5 解决冷启动问题时算法的新颖度对比

实验 2 中的新项目是在测试集中随机选取的,那么实验结果有可能会受随机选取数据的影响.因此,为了更进一步地比较本文提出 CUTA 推荐算法和 CUTATime 推荐算法的推荐效果,我们进行了实验 3.

实验 3. 解决冷启动问题:本文提出的 CUTA 算法和 CUTATime 算法的推荐准确度和新颖度对比.

首先,在实验 1 中进行实验的测试集中随机抽取 300 个、500 个、700 个、1 000 个项目依次作为新项目,训练集中对应的 300 个、500 个、700 个、1 000 个项目的评分信息及标签信息依次设为 0;然后,使用处理后的训练集和测试集对 CUTA 推荐算法和 CUTATime 推荐算法分别进行实验.同样地,部分实验结果记录在表 6 和表 7 中.其中,Prec 表示准确度,Nove 表示新颖度.

Table 6 Precisions and novelties for CUTA algorithm to solving cold-start problem

表 6 CUTA 算法的准确度和新颖度

近邻个数	本文个性化推荐算法 CUTA									
新项目个数 (α,β,γ)	160 (0.1,0.3,0.6)		300 (0.1,0.3,0.6)		500 (0.1,0.3,0.6)		700 (0.1,0.3,0.6)		1 000 (0.1,0.3,0.6)	
	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove
5	0.670	0.209	0.672	0.215	0.658	0.213	0.658	0.234	0.612	0.294
10	0.596	0.103	0.597	0.112	0.578	0.122	0.575	0.141	0.540	0.189
15	0.557	0.071	0.558	0.082	0.538	0.089	0.535	0.109	0.503	0.150
20	0.529	0.055	0.529	0.064	0.512	0.073	0.509	0.091	0.437	0.128
30	0.486	0.041	0.486	0.049	0.470	0.056	0.464	0.073	0.437	0.103

Table 7 Precisions and novelties for CUTATime algorithm to solving cold-start problem

表 7 CUTATime 算法准确度和新颖度

近邻个数	本文个性化推荐算法 CUTATime									
新项目个数 (α,β,γ)	160 (0.8,0.1,0.1)		300 (0.8,0.1,0.1)		500 (0.8,0.1,0.1)		700 (0.8,0.1,0.1)		1 000 (0.8,0.1,0.1)	
	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove
5	0.734	0.801	0.709	0.811	0.664	0.777	0.673	0.802	0.624	0.824
10	0.633	0.724	0.617	0.734	0.576	0.705	0.574	0.733	0.518	0.763
15	0.578	0.694	0.560	0.703	0.525	0.685	0.513	0.706	0.466	0.731
20	0.537	0.679	0.517	0.688	0.488	0.670	0.470	0.691	0.427	0.711
30	0.476	0.649	0.458	0.664	0.430	0.658	0.410	0.676	0.375	0.696

观察表 6 和表 7 可知:本文提出的 CUTA 推荐算法的推荐准确度与 CUTATime 推荐算法相差不多,但是 CUTATime 推荐算法的新颖度却远远超过 CUTA 推荐算法.综合考虑实验 2 和实验 3,证明了本文提出的 CUTATime 推荐算法在处理新项目冷启动问题上推荐效果最佳.

实验 4. 交叉验证本文提出的 CUTATime 个性化推荐算法.

为了进一步验证本文提出的 CUTATime 推荐算法的推荐效果,不受数据选取的随机性的影响,我们在实验 1 中进行实验的测试集中随机选取了 10 组包含 300 个新项目的数据集进行实验.同样地,我们如前在 36 种不同

参数取值情况下分别进行了实验,限于篇幅,表 8 只是给出了(α,β,γ)=(0.7,0.1,0.2)时的一组实验结果.

分析实验结果可知:本文提出的 CUTATime 推荐算法的推荐效果与数据选取随机性没有太大的影响;推荐准确度和新颖度的波动范围在 0.05 以内,可以接受.通过本次实验的交叉验证,更进一步地说明本文提出的 CUTATime 推荐算法解决了冷启动问题.

Table 8 Precisions and novelties for CUTATime algorithm by cross validation method

表 8 交叉验证 CUTATime 算法的准确度和新颖度

近邻个数	本文个性化推荐算法 CUTATime									
新项目个数 (α,β,γ)	300_1 (0.7,0.1,0.2)		300_2 (0.7,0.1,0.2)		300_3 (0.7,0.1,0.2)		300_4 (0.7,0.1,0.2)		300_5 (0.7,0.1,0.2)	
	Prec	Nove	Prec	Nove	Prec	Prec	Nove	Prec	Nove	Prec
5	0.691	0.807	0.646	0.807	0.650	0.832	0.679	0.793	0.654	0.811
10	0.691	0.748	0.552	0.735	0.557	0.768	0.569	0.723	0.561	0.737
15	0.530	0.722	0.506	0.714	0.493	0.739	0.509	0.704	0.503	0.714
20	0.486	0.701	0.468	0.699	0.448	0.728	0.466	0.698	0.466	0.697
30	0.424	0.682	0.412	0.675	0.388	0.707	0.407	0.681	0.412	0.683
新项目个数 (α,β,γ)	300_6 (0.7,0.1,0.2)		300_7 (0.7,0.1,0.2)		300_8 (0.7,0.1,0.2)		300_9 (0.7,0.1,0.2)		300_10 (0.7,0.1,0.2)	
	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove	Prec	Nove
5	0.546	0.743	0.555	0.757	0.562	0.745	0.544	0.735	0.554	0.753
10	0.499	0.716	0.497	0.734	0.506	0.723	0.493	0.708	0.501	0.727
15	0.462	0.698	0.457	0.711	0.467	0.709	0.455	0.695	0.461	0.709
20	0.411	0.678	0.404	0.683	0.414	0.686	0.400	0.678	0.405	0.686
30	0.643	0.803	0.659	0.814	0.674	0.803	0.639	0.800	0.656	0.812

5 总 结

由于互联网用户能够主动参与网络信息,使得网络信息更加丰富,信息数量也是与日俱增,进而使得网络信息过载问题受到各界研究学者的广泛关注;同时,作为有效解决信息过载问题的个性化推荐技术也得到了迅速发展.但是个性化推荐算法仍然面临着许多挑战,例如实时性、数据稀疏性、冷启动等问题.

本文研究了推荐算法中的新项目冷启动问题.首先,在分析用户评分、项目标签、项目属性信息的基础上定义了基于用户信息的预测评分值、基于项目标签信息的预测评分值和基于项目属性信息的预测评分值的计算公式,从而提出多种个性化推荐算法以提高推荐系统的推荐准确度,并进行了分析与实验验证;然后,结合项目属性和用户评价时间信息提出解决新项目冷启动问题的 CUTATime 个性化推荐算法,其中,项目属性和用户评价时间信息弥补了新项目开始加入时不存在评分和标签信息的缺陷,并且可以从另外一个角度直接反映用户对项目的偏好程度,实现对不同用户、不同偏好的个性化推荐;最后,通过多组实验对比和交叉验证,表明本文提出的 CUTATime 个性化推荐算法不仅推荐准确度高,而且有效地解决了新项目冷启动问题.

在该研究工作中,如果能够更多地考虑针对每个项目而言的用户评价时间与项目发布时间的差值,会更好地反映用户对于某项目(可以扩展到某类项目)的偏好,将有利于用户兴趣模型工作的建立;此外,如何考虑一个合适的部分数据来指导运算(推荐),对于推荐系统的实时性非常重要.如果能够建立用户的动态兴趣模型,将有助于该问题的解决.另一方面,我们可以通过分析用户之间动态变化的社交信息,根据用户之间的联系程度对用户进行社区划分,将社区内的相似用户喜欢的项目推荐给刚加入到社区内的新用户,解决推荐算法中的新用户冷启动问题.

References:

[1] Huang LW, Li DY. A review of information recommendation in social media. CAAI Trans. on Intelligent Systems, 2012,7(1):1−8 (in Chinese with English abstract).

[2] Wang LC, Meng XW, Zhang YJ. Context-Aware recommender systems: A survey of the state-of-the-art and possible extensions. Ruan Jian Xue Bao/Journal of Software, 2012,23(1):1−20 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4100. htm [doi: 10.3724/SP.J.1001.2012.04100]

[3] Deshpande M, Karypis G. Item-Based top-*N* recommendation algorithms. ACM Trans. on Information Systems, 2004,22(1): 143−177. [doi: 10.1145/963770.963776]

[4] Xu HL, Wu X, Li XD, Yan BP. Comparison study of internet recommendation system. Ruan Jian Xue Bao/Journal of Software, 2009, 20(2):350−362 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3388.htm

[5] Park ST, Chu W. Pairwise preference regression for cold-start recommendation. In: Proc. of the 3rd ACM Conf. on Recommender Systems. ACM Press, 2009. 21−28. [doi: 10.1145/1639714.1639720]

[6] Zhu R, Wang HM, Feng DW. Trustworthy services selection based on preference recommendation. Ruan Jian Xue Bao/Journal of Software, 2011,22(5):852−864 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3804.htm [doi: 10.3724/SP.J.1001.2011.03804]

[7] Luo XJ, Wang TC, Du XY, Liu HY, He J. Recommendation based on category—A method to solve cold-start problem in collaborative filtering. Journal of Computer Research and Development, 2007,44(Suppl.):290−295 (in Chinese with English abstract).

[8] Li G, Li L. A new algorithm of cold-start in a collaborative filtering system. Journal of Shandong University (engineering science), 2012,2(24):12−17 (in Chinese with English abstract).

[9] Zhang ZK, Liu C, Zhang YC, Zhou T. Solving the cold-start problem in recommender systems with social tags. EPL (Europhysics Letters), 2010,92(2):28002. [doi: 10.1209/0295-5075/92/28002]

[10] Zhang ZK, Zhou T, Zhang YC. Tag-Aware recommender systems: A state-of-the-art survey. Journal of Computer Science and Technology, 2011,26(5):767−777. [doi: 10.1007/s11390-011-0176-1]

[11] Marinho LB, Nanopoulos A, Thiemel S. Social tagging recommender systems. In: Ricci F, Rokach L, Shapira B, ed. Recommender Systems Handbook. New York: Springer-Verlag, 2011. 615−644. [doi: 10.1007/978-0-387-85820-3_19]

[12] Mistry O, Sen S. Tag recommendation for social bookmarking: Probabilistic approaches. Multiagent and Grid Systems, 2012,8(2): 143−163. [doi: 10.3233/MGS-2012-0190]

[13] Jomsri P, Sanguansintukul S, Choochaiwattana W. A framework for tag-based research paper recommender system: An IR approach. In: Proc. of the 2010 IEEE 24th Int'l Conf. on Advanced Information Networking and Applications Workshops. 2010. 103−108. [doi: 10.1109/WAINA.2010.35]

[14] Zhang ZK, Zhou T, Zhang YC. Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs. Physica A: Statistical Mechanics and its Applications, 2010,389(1):179−186. [doi: 10.1016/j.physa.2009.08.036]

[15] Koren Y. Collaborative filtering with temporal dynamics. In: Proc. of the 15th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Paris, 2009. 447−456. [doi: 10.1145/1557019.1557072]

[16] Xiang L, Yuan Q, Zhao SW, Chen L, Zhang XT, Yang Q, Sun J. Temporal recommendation on graphs via long- and short-term preference fusion. In: Proc. of the 16th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2010. 723−732. [doi: 10.1145/1835804.1835896]

[17] Zheng N, Li QD. A recommender system based on tag and time information for social tagging systems. Expert Systems with Applications, 2011,38(4):4575−4587. [doi: 10.1016/j.eswa.2010.09.131]

[18] Huang CG, Yin J, Wang J, Liu YB, Wang JH. Uncertain neighbors collaborative filtering recommendation algorithm. Chinese Journal of Computers, 2010,8(33):1370−1377 (in Chinese with English abstract).

[19] Zhang F, Sun XD, Chang HY, Zhao JS. Research on privacy-preserving two-party collaborative filtering recommendation. Acta Electronica Sinica, 2009,37(1):84−89 (in Chinese with English abstract).

[20] Chou AY. The analysis of online social networking: How technology is changing e-commerce purchasing decision. Int'l Journal of Information Systems and Change Management, 2010,4(4):353−365. [doi: 10.1504/IJISCM.2010.036917]

[21] MovieLens data sets. 2012. http://files.grouplens.org/datasets/hetrec2011/hetrec2011-movielens-2k-v2.zip

[22] Su JH, Yeh HH, Yu PS, Tseng VS. Music recommendation using content and context information mining. IEEE Intelligent Systems, 2010, 25(1):16−26. [doi: 10.1109/MIS.2010.23]

附中文参考文献:

[1] 黄立威,李德毅.社交媒体中的信息推荐.智能系统学报,2012,7(1):1−8.

[2] 王立才,孟祥武,张玉洁.上下文感知推荐系统.软件学报,2012,23(1):1−20. http://www.jos.org.cn/1000-9825/4100.htm [doi: 10.3724/SP.J.1001.2012.04100]

[4] 许海玲,吴潇,李晓东,阎保平.互联网推荐系统比较研究.软件学报,2009,20(2):350−362. http://www.jos.org.cn/1000-9825/3388.htm

[6] 朱锐,王怀民,冯大为.基于偏好推荐的可信服务选择.软件学报,2011,22(5):852−864. http://www.jos.org.cn/1000-9825/3804.htm [doi: 10.3724/SP.J.1001.2011.03804]

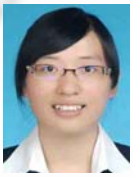
[7] 罗喜军,王韬丞,杜小勇,刘红岩,何军.基于类别的推荐——一种解决协同过滤推荐中的冷启动问题的方法.计算机研究与发展,2007,44(Suppl.):290−295.

[8] 李改,李磊.一种解决协同过滤系统冷启动问题的新算法.山东大学学报(工学版),2012,2(24):12−17.

[18] 黄创光,印鉴,汪静,刘玉葆,王甲海.不确定近邻的协同过滤推荐算法.计算机学报,2010,8(33):1370−1377.

[19] 张锋,孙雪冬,常会友,赵淦森.两方参与的隐私保护协同过滤推荐研究.电子学报,2009,37(1):84−89.

于洪(1972－),女,重庆人,博士,教授,CCF 会员,主要研究领域为粗糙集理论,Web 智能,人工智能,智能推荐.



李俊华(1987－),女,硕士,主要研究领域为 Web 智能,数据挖掘.