结合类别偏好信息的 Item-based 协同过滤算法*

冷亚军[1]，陆 青[1]，张俊岭[2]
（1. 上海电力学院 经济与管理学院，上海 201300；2. 浙江师范大学 经济与管理学院，浙江 金华 321004）

摘 要：传统的基于项目的协同过滤算法离线计算项目相似性，提高了在线推荐速度，但该算法仍然不能解决数据稀疏性所带来的问题，计算出的项目相似性准确度较差，影响了推荐质量。针对这一问题，提出了一种结合类别偏好信息的协同过滤算法。定义了类别偏好相似性，采用类别偏好相似性方法为目标项目找出一组类别偏好相似的候选邻居，在候选邻居中搜寻最近邻，排除了与目标项目共同评分较少项目的干扰，从整体上提高了最近邻搜寻的准确性。在 MovieLens 数据集上进行了实验，实验结果表明，新算法的推荐质量较传统的基于项目的协同过滤算法有显著提高。

关键词：推荐系统；协同过滤；类别偏好；相似性
中图分类号：TP181 **文献标志码**：A **文章编号**：1001-3695(2016)03-669-04
doi:10.3969/j.issn.1001-3695.2016.03.007

Improved item-based collaborative filtering algorithm combined with class preference information

Leng Yajun[1], Lu Qing[1], Zhang Junling[2]
(1. *College of Economics & Management, Shanghai University of Electric Power, Shanghai* 201300, *China*; 2. *School of Economics & Management, Zhejiang Normal University, Jinhua Zhejiang* 321004, *China*)

Abstract: The traditional item-based collaborative filtering(CF) algorithm computes item-item similarity offline, so it enhances the real-time performance of recommender system. However, item-based CF algorithm still suffers from the data sparsity problem, as a result that the recommendation quality is poor. To address this issue, this paper proposed a novel CF algorithm combined with class preference information. The proposed algorithm first found out candidate neighbors who were similar to the target item in class preference. Then it searched for nearest neighbors in the candidate neighbor set, which eliminated the interference of the items those had few co-ratings with the target item. Experimental results based on MovieLens dataset show that the recommendation quality of the new algorithm is significantly improved compared with traditional item-based CF algorithm.
Key words: recommender system; collaborative filtering; class preference; similarity

0 引言

推荐系统(recommender system)根据用户的偏好向用户提供个性化的信息、商品和服务的推荐，帮助用户解决信息超载(information overload)问题带来的困扰。随着互联网上信息的增长和用户个性化需求的提高，推荐系统的应用日益广泛，成为电子商务、社会网络、视频/音乐点播等主流 Web 2.0 服务的核心技术[1,2]。协同过滤(collaborative filtering)是推荐系统中广泛使用的最成功的推荐技术[3]，它根据相似用户的偏好推测目标用户的偏好，向目标用户推荐其最有可能感兴趣的资源项(如 Web 页面、音乐、电影、商品等)。协同过滤技术的优势在于不需要考虑被推荐资源项的内容，并且可以产生新异的推荐。尽管协同过滤技术在个性化推荐方面取得了很大成功，但是随着系统中用户数目和商品数目的不断增加，该技术面临着严峻的挑战，主要有两方面[4]：a)缺乏可扩展性，即数据规模巨大，导致推荐算法计算复杂度急剧增加，无法及时产生推荐；b)数据稀疏性，实际的系统中资源项数量庞大，而用户通常只对一小部分资源项进行评分，可利用的表示用户偏好的信息非常有限。

针对协同过滤中的可扩展性问题，研究人员陆续提出了很多解决方法，包括数据降维的模型，如奇异值分解(singular value decomposition, SVD)[5]、正交非负矩阵分解(orthogonal nonnegative matrix tri-factorization, ONMTF)[6]等；基于聚类技术的模型，如基于用户聚类(user clustering)[7]的方法、联合聚类(co-clustering)[8]方法等；基于项目的协同过滤模型(item-based collaborative filtering)[9]。基于项目的协同过滤离线计算项目的相似性，在线时通过调用事先计算好的相似性数据，在 $O(n)$ 时间内找到目标项目的 k 个最相似项目并产生推荐(n 是系统中项目总数)，提高了推荐速度。离线计算项目相似性的基本依据是项目间的相似性比较稳定，一定时期内不会有太

收稿日期：2015-10-26；**修回日期**：2015-12-03 **基金项目**：国家自然科学基金资助项目(71201145)；上海市教育委员会科研创新资助项目(15ZS064)；上海电力学院科研基金资助项目(K2014-037)

作者简介：冷亚军(1985-)，男，辽宁铁岭人，讲师，博士，主要研究方向为电子商务、数据挖掘(huayi2001@163.com)；陆青(1982-)，男，上海人，讲师，博士，主要研究方向为进化计算、数据挖掘；张俊岭(1981-)，男，安徽合肥人，副教授，博士，主要研究方向为智能决策支持系统、进化算法.

大变化。然而,基于项目的协同过滤仍然不能解决数据稀疏性所带来的问题,该方法计算出的项目相似性准确度较差,影响了推荐质量。基于此,本文提出一种新的基于项目的协同过滤算法,找出与目标项目类别偏好相似的候选邻居项目,在候选邻居中搜寻最近邻,排除与目标项目共同评分较少项目的干扰,从整体上提高算法的推荐质量。

本文的贡献包括三个方面:a)提出了一种新的相似性概念——类别偏好相似性,揭示了项目共同评分数与类别偏好相似性之间的关系;b)提出了一种改进的基于项目的协同过滤算法,该算法综合考虑了项目间类别偏好相似性和评分相似性的作用,具有更高的推荐质量;c)在真实数据集上对算法性能进行了检验。

1 相关研究

1.1 基于项目的协同过滤算法

基于项目的协同过滤算法在用户—项目评分矩阵 $R(m,n)$ 上进行计算,m 表示用户数,n 表示项目数,矩阵中每一实体 $R_{i,j}$ 表示用户 i 对项目 j 的评分值,体现了用户 i 对项目 j 的偏好。基于项目的协同过滤算法计算目标项目 i 与其他项目的相似性,选择相似性最高的前 k 个项目组成 i 的最近邻 $I_n=\{i_1,i_2,\cdots,i_k\}$。项目相似性采用表 1 中的方法进行度量。

表 1 项目相似性度量方法[9,10]

measures	description
pearson correlation coefficient	$\mathrm{sim}(i,j)=\dfrac{\sum_{u\in U_{ij}}(R_{u,i}-\overline{R_i})(R_{u,j}-\overline{R_j})}{\sqrt{\sum_{u\in U_{ij}}(R_{u,i}-\overline{R_i})^2}\sqrt{\sum_{u\in U_{ij}}(R_{u,j}-\overline{R_j})^2}}$
cosine similarity	$\mathrm{sim}(i,j)=\cos(i,j)=\dfrac{i\times j}{\|i\|_2\times\|j\|_2}=\dfrac{\sum_{u\in U_{ij}}R_{u,i}\times R_{u,j}}{\sqrt{\sum_{u\in U_{ij}}R_{u,i}^2}\sqrt{\sum_{u\in U_{ij}}R_{u,j}^2}}$
adjusted cosine similarity	$\mathrm{sim}(i,j)=\dfrac{\sum_{u\in U_{ij}}(R_{u,i}-\overline{R_u})(R_{u,j}-\overline{R_u})}{\sqrt{\sum_{u\in U_{ij}}(R_{u,i}-\overline{R_u})^2}\sqrt{\sum_{u\in U_{ij}}(R_{u,j}-\overline{R_u})^2}}$

表 1 中,i、j 表示项目空间中两个项目;$U_{i,j}$ 表示对 i、j 都有评分的用户集合;$R_{u,i}$ 表示用户 u 对项目 i 的评分;$\overline{R_i}$ 表示项目 i 的平均评分;$\overline{R_u}$ 表示用户 u 的平均评分。

最近邻产生后,对于用户 u 尚未评分的项目 i,按式(1)加权计算得到 u 对 i 的预测评分 $P_{u,i}$,将预测评分最高的前 n 个项目推荐给 u。

$$P_{u,i}=\overline{R_i}+\frac{\sum_{i_k\in I_n}\mathrm{sim}(i,i_k)\times(R_{u,i_k}-\overline{R_{i_k}})}{\sum_{i_k\in I_n}|\mathrm{sim}(i,i_k)|}\tag{1}$$

1.2 局限性

基于项目的协同过滤算法在两个项目的共同评分集合上计算项目的相似性,只有当两个项目获得的共同评分较多时,计算出的相似性才比较可信。实际上,由于评分数据极端稀疏,两个项目获得的共同评分很少,通常只有一两对,即使项目在这样小的共同评分集合上获得的评分非常相似,也不能确定它们之间具有很高的相似性。基于项目的协同过滤算法在整个项目空间搜寻最近邻,很多与目标项目共同评分较少、原本与目标项目差别较大的项目,由于计算出的评分相似性较高被选为最近邻,这在一定程度上降低了最近邻搜寻的准确性,影响了算法的推荐质量。

2 结合类别偏好信息的协同过滤推荐算法

2.1 类别偏好

设系统中的用户集合为 $U=\{u_1,u_2,\cdots,u_m\}$。采用聚类技术把用户划分为 t 类即 $C=\{c_1,c_2,\cdots,c_t\}$,每个类中包含的用户尽可能相似,两类间的用户尽可能不同,并有如下性质:$U=c_1\cup c_2\cup\cdots\cup c_t$,$c_i\cap c_j=\varnothing(1\leqslant i\leqslant t,1\leqslant j\leqslant t)$。虽然系统中用户数量庞大,但是一个项目通常只会获得一小部分用户的评分,且这些评分往往集中在一个或少数几个用户类中,因此项目在这些用户类上获得的评分相对稠密。项目所获评分在用户类上的分布反映了项目在各用户类中受到的偏好程度。由表 2 可见,项目 i_1 的评分集中在类 c_1 中,表明 i_1 更多受到 c_1 的偏好;项目 i_2 的评分集中在 c_2 和 c_3 中,项目 i_3 的评分集中在 c_1 和 c_3 中,相对于 i_2,i_1 与 i_3 受到的用户类别偏好更加相似。

表 2 评分数据分布在不同用户类

项	c_1					c_2					c_3				
目	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}	u_{15}
i_1	4	3	2	5						2					1
i_2						5	3	3	4	2		3	3	5	1
i_3	3	2	4	4	5						4	3	3	4	3

本文算法首先为目标项目寻找一组类别偏好相似的候选邻居集合,这些候选邻居与目标项目受到相同用户类的偏好,因此与目标项目共同评分较多。例如,从 MovieLens100K 数据集[11](详见 3.1 节)中随机抽取 100 个项目作为目标项目,统计这些项目与邻居项目的共同评分数随用户类别偏好相似性的变化。从图 1 可以看出,随着类别偏好相似性的不断下降,目标项目与邻居项目间的共同评分数逐渐减少。表 3 给出了详细数据,类别偏好相似性最高的前 20 个项目与目标项目的平均共同评分数为 38.0,而类别偏好相似性排序为 1001 ~ 1020 的项目与目标项目的平均共同评分数则仅为 5.3。由此可得出结论:项目共同评分数与类别偏好相似性成正比关系,项目间类别偏好相似性越高,项目获得的共同评分数往往越多。

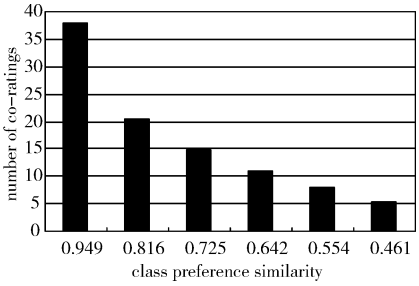

图 1 共同评分数随类别偏好相似性的变化

表 3 详细数据

与目标项目类别偏好相似性排序	平均类别偏好相似性	共同评分总数	平均共同评分数
1 ~ 20	0.949	75 992	38.0
201 ~ 220	0.816	40 627	20.3
401 ~ 420	0.725	29 769	14.9
601 ~ 620	0.642	21 822	10.9
801 ~ 820	0.554	16 121	8.1
1001 ~ 1020	0.461	10 646	5.3

2.2 确定候选邻居

定义 1 类别偏好值。设 U 表示整个用户空间,C 表示经

过聚类生成的用户类集合,j 为项目空间中任一项目,$U_j = \{u \in U | R_{u,j} \neq \varnothing\}$ 表示对项目 j 进行过评分的用户集合,$\forall c_s \in C$,则称 $P_{j,s} = |c_s \cap U_j| / |U_j| (0 \leqslant P_{j,s} \leqslant 1)$ 为项目 j 在 c_s 上的类别偏好值。

根据类别偏好值本文构造出项目—类别偏好矩阵 $P(n,t)$,如表 4 所示。n 行表示 n 个项目,t 列表示 t 个用户类,$P_{j,s}$ 表示项目 j 在用户类 s 上的偏好值。在 $P(n,t)$ 上计算目标项目 i 与其他项目的类别偏好相似性,按照类别偏好相似性从高到低的顺序排列所有项目,组成 i 的候选邻居列表。

表 4　项目—类别偏好矩阵 $P(n,t)$

	c_1	…	c_s	…	c_t
$item_1$	$P_{1,1}$	…	$P_{1,s}$	…	$P_{1,t}$
…	…	…	…	…	…
$item_j$	$P_{j,1}$	…	$P_{j,s}$	…	$P_{j,t}$
…	…	…	…	…	…
$item_n$	$P_{n,1}$	…	$P_{n,s}$	…	$P_{n,t}$

2.3　结合类别偏好信息的协同过滤算法及性能分析

本文提出的结合类别偏好信息的协同过滤算法分为离线处理和在线推荐两个阶段。离线时首先构造项目—类别偏好矩阵 $P(n,t)$,在 $P(n,t)$ 上计算项目类别偏好相似性,生成项目候选邻居列表 T_{cn};然后计算项目与候选邻居的评分相似性,生成项目最近邻列表 T_{nn}。在线阶段,当用户到达时,根据 T_{nn} 及事先计算好的项目相似性预测用户对未评分项目的评分。算法具体流程如图 2 所示。

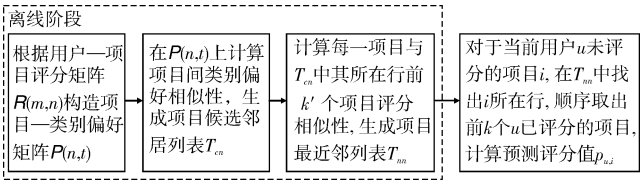

图 2　本文算法流程

算法 1　结合类别偏好信息的协同过滤推荐算法 CPCF

输入:用户—项目评分矩阵 $R(m,n)$,目标用户 u,待评分项目 i,用户类别数 t,候选邻居数 k',最近邻数 k。

输出:u 对 i 的预测评分。

过程如下:

a)根据 K-means 聚类算法[12] 把用户集 U 划分为 t 类,得到用户类集合 $C = \{c_1, c_2, \cdots, c_t\}$。

b)在 $R(m,n)$ 上计算出项目 j 的已评分的用户集合 $U_j = \{u \in U | R_{u,j} \neq \varnothing\}$。

c)$\forall c_s \in C$,计算项目 j 在 c_s 上的类别偏好值 $P_{j,s} = |c_s \cap U_j| / |U_j|$。

d)循环执行 b)、c),得到每个项目在各用户类上的偏好值,构造项目—类别偏好矩阵 $P(n,t)$。

e)在 $P(n,t)$ 上计算项目间类别偏好相似性,生成项目候选邻居列表 T_{cn} 并保存。

f)在 $R(m,n)$ 上计算每一项目与 T_{cn} 中前 k' 个候选邻居的评分相似性,生成项目最近邻列表 T_{nn} 并保存。

g)在 T_{nn} 中找到 i 所对应的行,顺序取出前 k 个 u 已评分的项目。

h)根据式(1)计算 u 对 i 的预测评分 $P_{u,i}$。

本文算法分为离线处理 a)~f)和在线推荐 g)h)两个阶段。由于系统中项目性质比较稳定,并且项目的增加速度相对于用户而言要慢很多,所以算法 1 中 a)~f)只需定期离线计算一次即可,并不影响推荐速度。离线时首先计算目标项目与其他项目的类别偏好相似性,确定候选邻居,计算复杂度为 $O(t \times n)$,其中 t 表示用户类数,n 表示项目数;然后在候选邻居中搜寻目标项目的最近邻,计算复杂度为 $O(m \times k')$,其中 m 表示用户数,k' 表示候选邻居数。影响推荐速度的是步骤 g)h),g)顺序取出目标用户已评分的前 k 个项目,计算复杂度最坏情况为 $O(k')$;步骤 h)通过加权 k 个最近邻的评分产生对目标项目的预测评分,计算复杂度为 $O(k)$。

3　实验结果及分析

3.1　数据集

本文采用 MovieLens100K 数据集[11] 对算法进行评估。MovieLens100K 数据集包含 943 位用户对 1 682 部电影的 100 000条评分记录(评分值为 1~5 的整数),数据集的稀疏等级为 1 - 100000/(943 × 1682) = 0.9370。从该数据集随机抽取 600 位用户的评分数据组成本文实验数据集,基本特征如表 5 所示。实验中将数据集按照 80%/20% 的比例划分为训练集和测试集。

表 5　实验数据集基本特征

	用户总数	项目总目	评分总数	用户最大评分数	用户最小评分数	用户平均评分数	稀疏等级
实验数据集	600	1 625	66 277	377	20	110.462	0.932 0

3.2　评价标准

本文实验采用平均绝对偏差 MAE(mean absolute error)作为度量算法优劣的标准[13,14]。MAE 通过计算用户的预测评分与实际评分之间的偏差来度量预测的准确性,MAE 值越小,推荐质量越高。设测试集中共有 H 条评分数据,分别为 $\{q_1, q_2, \cdots, q_H\}$,算法对这些数据的预测值为 $\{p_1, p_2, \cdots, p_H\}$,则算法的 MAE 为

$$\mathrm{MAE} = \frac{\sum_{i=1}^{H} |p_i - q_i|}{H} \tag{2}$$

3.3　实验结果

实验探讨各参数对本文算法的影响,在最优参数基础上对本文算法和传统协同过滤算法作比较。相似性度量方法采用 Pearson 相关系数。

1)用户类别数和候选邻居数

首先考察用户类别数对本文算法的影响,将用户聚为10~80 类,固定候选邻居数 $k' = 500$。如图 3 所示,当用户类别数 t 取 50 时,算法具有最小的 MAE 值,以后的实验取 $t = 50$。接下来考察候选邻居数的影响,将用户聚为 50 类,候选邻居数 k' 在 100~800 变动。如图 4 所示,k' 取 200 时,算法的 MAE 值达到最小;随后再增加 k' 值,算法的 MAE 值增大,推荐质量下降。因此,本文取 $k' = 200$ 进行后续的实验。

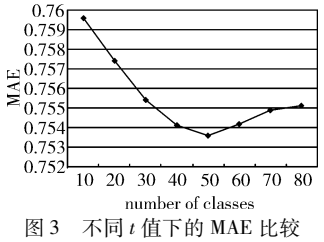

图 3　不同 t 值下的 MAE 比较

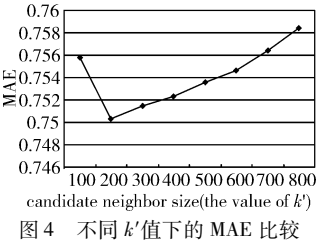

图 4　不同 k' 值下的 MAE 比较

2)相似性调整系数

节计算目标项目与候选邻居的评分相似性 sim_R,选择 sim_R 较大的项目作为目标项目的最近邻,忽视了项目间类别偏好相似性 sim_P 的作用。这里综合考虑了 sim_R 和 sim_p,根据式(3)计算目标项目与候选邻居的综合相似性,选择综合相似性最大的前 k 个项目作为最近邻,生成预测评分。

$$sim_I = \alpha\, sim_P + (1-\alpha) sim_R \tag{3}$$

其中:sim_I 表示项目综合相似性;α 为[0,1]内的任意实数,用于调节 sim_R 和 sim_P 所占比重。

实验中 α 从 0.1 增加到 0.9,每次递增 0.1。如图 5 所示,随着 α 的不断增大,算法的 MAE 逐渐降低,当 α 取 0.4 时,算法取得最低的 MAE;随后再增加 α 值,算法的 MAE 值增大。

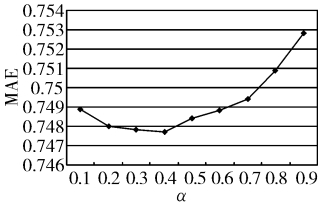

图 5 不同 α 值下的 MAE 比较

3)与传统协同过滤算法的比较

将本文算法和传统的基于用户的协同过滤算法[15](记为 UBCF)、基于项目的协同过滤算法[9](记为 IBCF)作比较。实验中,分别取 α 为 0 和 0.4,前者仅采用评分相似性进行计算(记为 CPRS);后者综合考虑了评分相似性和类别偏好相似性的作用(记为 CPIS)。

变化最近邻数 k,比较各算法的 MAE,结果如图 6 所示。从图 6 可以看出,在任意 k 值下 CPRS、CPIS 都取得了比 UBCF 和 IBCF 更低的 MAE;且综合考虑 sim_R 和 sim_p 的 CPIS,在四种算法中具有最高的推荐质量。为了检验本文算法与传统协同过滤算法的推荐准确性是否存在显著差异,对算法进行了方差分析[16]。如表 6 所示,各算法比较后的 P-value 均小于 0.05,表明本文算法 CPRS、CPIS 的推荐准确性比 UBCF 和 IBCF 有显著提高。

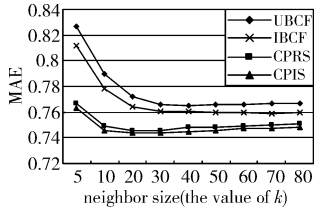

图 6 不同算法的 MAE 比较

表 6 方差分析(显著性水平为 0.05)

算法	CPRS vs. UBCF	CPRS vs. IBCF	CPIS vs. UBCF	CPIS vs. IBCF
P-value	0.002 5	0.010 6	0.001 1	0.004 0

4 结束语

推荐系统在向网络用户提供个性化服务方面发挥着至关重要的作用,目前许多大型网站都应用了推荐系统。基于项目的协同过滤由于离线计算项目的相似性,提高了推荐实时性,成为推荐系统中广泛使用的推荐技术。然而该算法仍然不能解决数据稀疏性带来的问题,推荐质量较差。本文提出了一种结合类别偏好信息的协同过滤算法,首先找出与目标项目类别偏好相似的候选邻居,候选邻居与目标项目性质相近,共同评分较多;在候选邻居中搜寻最近邻,排除了与目标项目共同评分较少项目的干扰,从整体上提高了最近邻搜寻的准确性,从而提高了推荐质量。在 MovieLens100K 数据集上进行了实验,考察了用户类别数、候选邻居数和相似性调整系数对本文算法的影响,比较了本文算法和传统协同过滤算法的推荐质量。实验结果表明,本文算法的推荐质量较传统的协同过滤算法有显著提高。本文算法对用户有显著群体特征的系统具有很好的推荐效果,但本文实验所用数据集为电影评分数据集,数据集中的用户没有显著群体划分,而是通过聚类算法模拟真实群体特征数据集。未来的工作尝试将本文算法应用于真实的群体特征数据集,提高算法的应用价值。

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] 吴湖,王永吉,王哲,等. 两阶段联合聚类协同过滤算法[J]. 软件学报, 2010, 21(5): 1042-1054.

[3] Al-Shamri M Y H. Power coefficient as a similarity measure for memory-based collaborative recommender systems [J]. Expert Systems with Applications, 2014, 41(13): 5680-5688.

[4] Choi K, Suh Y. A new similarity function for selecting neighbors for each target item in collaborative filtering [J]. Knowledge-Based Systems, 2013, 37(1): 146-153.

[5] Vozalis M G, Margaritis K G. Using SVD and demographic data for the enhancement of generalized collaborative filtering [J]. Information Sciences, 2007, 177(15): 3017-3037.

[6] Chen Gang, Wang Fei, Zhang Changshui. Collaborative filtering using orthogonal nonnegative matrix tri-factorization [J]. Information Processing and Management, 2009, 45(3): 368-379.

[7] Sarwar B M, Karypis G, Konstan J, *et al.* Recommender systems for large-scale e-commerce: scalable neighborhood formation using clustering [C]//Proc of the 5th International Conference on Computer and Information Technology. 2002.

[8] Shan H H, Banerjee A. Bayesian co-clustering [C]//Proc of IEEE International Conference on Data Mining. Washington: IEEE Computer Society, 2008: 530-539.

[9] Sarwar B M, Karypis G, Konstan J A, *et al.* Item-based collaborative filtering recommendation algorithms [C]//Proc of the 10th International Conference on World Wide Web. New York: ACM Press, 2001: 285? 295.

[10] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions [J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17(6): 734-749.

[11] GroupLens[EB/OL]. http://www.grouplens.org/.

[12] 朱明. 数据挖掘[M]. 合肥:中国科学技术大学出版社, 2002.

[13] Yang Xiwang, Guo Yang, Liu Yong, *et al.* A survey of collaborative filtering based social recommender systems [J]. Computer Communications, 2014, 41(3): 1-10.

[14] 梁昌勇,冷亚军,王勇胜,等. 电子商务推荐系统中群体用户推荐问题研究[J]. 中国管理科学, 2013, 21(3): 153-158.

[15] Resnick P, Iacovou N, Suchak M, *et al.* GroupLens: an open architecture for collaborative filtering of netnews [C]//Proc of ACM Conference on Computer-Supported Cooperative Work. New York: ACM Press, 1994: 175-186.

[16] Thompson B. The concept of statistical significance testing, practical assessment [J]. Research & Evaluation, 1994, 4(5): 5-6.