计算机研究与发展 ISSN 1000-1239/CN 11-1777/TP
Journal of Computer Research and Development 45(9): 1532-1538, 2008

基于领域最近邻的协同过滤推荐算法

李 聪[1] 梁昌勇[1] 马 丽[2]
[1](合肥工业大学管理学院 合肥 230009)
[2](西华师范大学商学院 四川南充 637002)
(cnlicong@yahoo.cn)

A Collaborative Filtering Recommendation Algorithm Based on Domain Nearest Neighbor

Li Cong[1], Liang Changyong[1], and Ma Li[2]
[1](*School of Management, Hefei University of Technology, Hefei* 230009)
[2](*Business College, China West Normal University, Nanchong, Sichuan* 637002)

Abstract Currently E-commerce recommender systems are being used as an important business tool by an increasing number of E-commerce websites to help their customers find products to purchase. Collaborative filtering is the most successful and widely used recommendation technology in E-commerce recommender systems. However, traditional collaborative filtering algorithm faces severe challenge of sparse user ratings and real-time recommendation. To solve the problems, a collaborative filtering recommendation algorithm based on domain nearest neighbor is proposed. The union of user rating items is used as the basis of similarity computing among users, and the non-target users are differentiated into two types that without recommending ability and with recommending ability. To the former users, user similarity will not be computed for improving real-time performance; to the latter users, "domain nearest neighbor" method is proposed and used to predict missing values in the union of user rating items when the users have common intersections of rating item classes with target user, and then the needed items space for missing values predicting can be reduced to the few common intersections. Thus the sparsity can be decreased and the accuracy of searching nearest neighbor can be improved. The experimental results show that the new algorithm can efficiently improve recommendation quality.

Key words collaborative filtering; recommendation algorithm; domain nearest neighbor; user similarity; MAE

摘 要 协同过滤是目前电子商务推荐系统中广泛应用的最成功的推荐技术,但面临严峻的用户评分数据稀疏性和推荐实时性挑战.针对上述问题,提出了基于领域最近邻的协同过滤推荐算法,以用户评分项并集作为用户相似性计算基础,将并集中的非目标用户区分为无推荐能力和有推荐能力两种类型;对于前一类用户不再计算用户相似性以改善推荐实时性,对于后一类用户则提出"领域最近邻"方法对并集中的未评分项进行评分预测,从而降低数据稀疏性和提高最近邻寻找准确性.实验结果表明,该算法能有效提高推荐质量.

关键词 协同过滤;推荐算法;领域最近邻;用户相似性;平均绝对误差

中图法分类号 TP311

收稿日期:2007-04-29;修回日期:2008-05-22

基金项目:国家自然科学基金项目(70771037);教育部科学技术研究重点基金项目(107067);高等学校博士学科点专项科研基金项目(20050359006)

随着 Internet 和电子商务的迅猛发展, 电子商务推荐系统[1] 被电子商务网站用做虚拟店员(virtual salespeople)向客户提供商品信息和建议, 帮助用户决定应该购买何种商品, 其作用主要表现在 3 个方面: 1)将电子商务网站浏览者转变为购买者; 2)提高电子商务网站交叉销售能力; 3)建立客户忠诚度. 协同过滤(collaborative filtering)作为目前电子商务推荐系统中广泛使用的最成功的推荐算法[2], 使用统计技术寻找与目标用户有相同或相似兴趣偏好的邻居用户, 根据邻居用户的评分来预测目标用户对商品项的评分值, 选择预测评分最高的前 N 项商品作为推荐集反馈给目标用户, 其基本思想是用户会对邻居用户偏好的商品产生兴趣, 即基于用户(user-based)的协同过滤. 因此, 用户评分数据收集越多协同过滤算法的推荐质量越高. 但是随着电子商务站点用户和商品项数量的不断增加, 协同过滤面临严峻的用户评分数据稀疏性和推荐实时性挑战[1], 导致推荐质量迅速下降.

对此, 研究人员陆续提出了一些改进方法, 例如基于项目的协同过滤及其改进算法[2-4]、基于矩阵降维的协同过滤[5-7]、基于神经网络的协同过滤[8] 等等. 文献[9] 通过用户评分项并集来计算用户相似性, 以降低评分数据稀疏性. 本文进一步将用户评分项并集中的非目标用户区分为无推荐能力和有推荐能力两种类型, 对于前一类用户不再计算其与目标用户的相似性以改善推荐实时性, 对于后一类用户则提出基于领域最近邻的协同过滤推荐算法, 采用"领域最近邻"对并集中的未评分项进行评分预测, 使得最近邻搜寻更加准确; 同时新算法避免了文献[9] 算法中不必要的计算耗费, 使推荐实时性得到改善. 实验结果表明, 新算法能有效提高推荐质量.

1　相关工作

1.1　传统的最近邻寻找

传统协同过滤推荐算法基于用户-项目评分矩阵 $\boldsymbol{R}(m, n)$ 寻找目标用户的最近邻(nearest neighbor)集合. $\boldsymbol{R}(m, n)$ 是一个 $m\times n$ 阶矩阵, 如图 1 所示, 其中 m 行表示 m 个用户, n 列表示 n 个项目, $R_{i,j}$ 表示用户 i 对项目 j 的评分值.

	$item_1$	…	$item_j$	…	$item_n$
$user_1$	$R_{1,1}$	…	$R_{1,j}$	…	$R_{1,n}$
⋮	⋮	⋮	⋮	⋮	⋮
$user_i$	$R_{i,1}$	…	$R_{i,j}$	…	$R_{i,n}$
⋮	⋮	⋮	⋮	⋮	⋮
$user_m$	$R_{m,1}$	…	$R_{m,j}$	…	$R_{m,n}$

Fig. 1　User-item ratings matrix $\boldsymbol{R}(m, n)$.

图 1　用户-项目评分矩阵 $\boldsymbol{R}(m, n)$

由于大型电子商务站点用户及商品项的数量庞大且不断增加, 使得 $\boldsymbol{R}(m, n)$ 成为高维矩阵; 同时用户给予评分的商品项很少, 通常在1%以下[3], 导致 $\boldsymbol{R}(m, n)$ 中的评分数据极端稀疏.

对于目标用户 u, 算法需要搜寻 u 的最近邻集合 $U=\{u_1, u_2, \cdots, u_K\}$, $u\notin U$ 且 u 与 U 中用户 u_k 之间的相似性 $sim(u, u_k)$ $(1\leqslant k\leqslant K)$ 由大到小排列. 最近邻数量 K 的值可直接给定或通过相似性阈值来确定; 也可将这两种方法结合, 即在相似性大于阈值的用户中择取相似性最大的前 K 个用户. 用户相似性度量方法主要有余弦相似性(cosine similarity)、Pearson 相关系数(Pearson correlation coefficient)、约束 Pearson 相关系数(constrained Pearson correlation coefficient)等, 具体计算方法如表 1 所示:

Table 1　Measures of User Similarity[10-11]

表 1　用户相似性度量方法[10-11]

Measures	Description
Cosine Similarity	$sim(u, v)=\cos(\boldsymbol{u}, \boldsymbol{v})=\dfrac{\boldsymbol{u}\cdot\boldsymbol{v}}{\|\boldsymbol{u}\|_2\times\|\boldsymbol{v}\|_2}=\dfrac{\sum_{i\in I_{uv}}R_{u,i}\cdot R_{v,i}}{\sqrt{\sum_{i\in I_{uv}}R_{u,i}^2}\ \sqrt{\sum_{i\in I_{uv}}R_{v,i}^2}}$
Pearson Correlation Coefficient	$sim(u, v)=\dfrac{\sum_{i\in I_{uv}}(R_{u,i}-R_u)\cdot(R_{v,i}-R_v)}{\sqrt{\sum_{i\in I_{uv}}(R_{u,i}-R_u)^2}\ \sqrt{\sum_{i\in I_{uv}}(R_{v,i}-R_v)^2}}$
Constrained Pearson Correlation Coefficient	$sim(u, v)=\dfrac{\sum_{i\in I_{uv}}(R_{u,i}-R_{med})\cdot(R_{v,i}-R_{med})}{\sqrt{\sum_{i\in I_{uv}}(R_{u,i}-R_{med})^2}\ \sqrt{\sum_{i\in I_{uv}}(R_{v,i}-R_{med})^2}}$

表 1 中的 u, v 表示用户空间中任意两位用户, $u\neq v$; $sim(u, v)$ 表示 u, v 之间的相似性; I_{uv} 表示 u, v 的共同评分项集, 即对于 $\forall i\in I_{uv}$ 有 $R_{u,i}\neq\varnothing$ 且 $R_{v,i}\neq\varnothing$; $\boldsymbol{u}, \boldsymbol{v}$ 表示 u, v 各自在 I_{uv} 上的评分向量;

$R_{u,i}$, $R_{v,i}$ 表示 u, v 各自对项目 i 的评分; $\bar{R}_u$, $\bar{R}_v$ 表示 u, v 在 I_{uv} 上的平均评分; R_{med} 表示推荐系统所采用的评分制中值.

1.2 基于项目评分预测的最近邻寻找

文献[9]采用目标用户 u 和用户 v 的评分项并集 I'_{uv} 来计算用户相似性. 设 u, v 的评分项集合分别为 I_u, I_v, 则 $I'_{uv}=I_u\cup I_v$. 对于 u, v 在 I'_{uv} 中的未评分项 i, 则通过寻找 i 的相似邻居项集合来进行评分预测, 使得 u, v 对 I'_{uv} 中所有项均有评分, 从而使用表 1 中的相似性度量方法计算 u, v 的相似性 $sim(u,v)$. 该算法能够使得用户之间的共同评分项较多, 有效解决了用户评分数据极端稀疏情况下传统相似性度量方法存在的不足, 提高了推荐质量.

但是, 评分项并集 I'_{uv} 中的用户 v 实际上可分为无推荐能力和有推荐能力两种类型, 而文献[9]未能区别对待, 导致在其算法中存在不必要的计算耗费. 另外, 该算法基于整个 $R(m,n)$ 来寻找未评分项的相似邻居项集合, 因此对于每个未评分项算法都将扫描全体项目空间(实际的电子商务站点商品项可能多达数十万种), 导致运行时间大幅增加, 对推荐实时性带来新的压力.

2 基于领域最近邻的协同过滤推荐算法

2.1 用户评分项并集分析

对于目标用户 u 和用户 v 各自的评分项集合 I_u, I_v 及两者的评分项并集 I'_{uv}(设为 5 分制评分), 有:

1) 若 $I_v\subset I_u$, 即对于 $\forall i\in I_v$, 都有 $i\in I_u$ 成立. 例如, 图 2(a)中用户 v 的所有评分项都已被用户 u 评价过, 因此 v 不可能向 u 推荐项目, 即相对于 u 而言 v 无推荐能力, 故无需计算 $sim(u,v)$.

2) 若 $I_v\not\subset I_u$, 即 $\exists i\in I_v$ 且 $R_{u,i}=\varnothing$. 此时考虑两种不同情况:

① 若 $R_{v,i}\leqslant R_{med}$, 表明用户 v 对项目 i 不喜欢或无喜好倾向, 因此即使 v 是 u 的最近邻也不会向 u 推荐 i(例如图 2(b)中 v 不会向 u 推荐 I_4), 故 v 仍为无推荐能力用户, 无需计算 $sim(u,v)$.

② 若 $R_{v,i}>R_{med}$, 表明用户 v 对项目 i 存在喜好, 因此当 v 是 u 的最近邻时将向 u 推荐 i(例如图 3 中 v 能够向 u 推荐 I_4), 即 v 属于有推荐能力用户, 可考虑计算 $sim(u,v)$.

	I_1	I_2	I_3	I_4	I_5	I_6
u	2	5	3	3	4	1
v		5			4	

(a)

	I_1	I_2	I_3	I_4	I_5	I_6
u	2	5	3		4	1
v	2	5	3	1	4	1

(b)

Fig. 2 v is the user without recommending ability. (a) $I_v\subset I_u$ and (b) $I_v\not\subset I_u$ and $R_{v,i}\leqslant R_{med}$.

图 2 v 是无推荐能力用户. (a) $I_v\subset I_u$; (b) $I_v\not\subset I_u$ 且 $R_{v,i}\leqslant R_{med}$

	I_1	I_2	I_3	I_4	I_5	I_6
u	2	5	3		4	1
v	2	5	3	5	4	1

Fig. 3 v is the user with recommending ability.

图 3 v 是有推荐能力用户

2.2 领域最近邻

设用户 v 相对于目标用户 u 是有推荐能力用户, 则可以基于领域最近邻进行 I'_{uv} 中未评分项的评分预测. 在实际的电子商务站点中, 所有商品项都是划分到有限的若干个项类中, 例如最大的中文网上书店当当网(www.dangdang.com)将图书分为文学、管理、计算机等多种类别. 由此, 设 I_u, I_v 中各个项目所属项类的集合分别为 C_u, C_v, 则 C_u 和 C_v 之间存在以下两种情况:

1) 若 $C_u\cap C_v=\varnothing$, 即 I_u, I_v 各自的项目分布在不同项类, 如表 2 所示:

	Literature			Management		Computer	
	I_1	I_2	I_3	I_4	I_5	I_6	I_7
u	2	5	1			5	3
v				4	5		

Fig. 4 The user ratings distributed over different item classes.

图 4 用户评分项分布在不同项类

从图 4 可见, 按文献[9]算法将需要填补较多的空缺评分值. 这种情况下容易产生准确性过低的预测评分而降低用户相似性计算质量, 且其采用的项目相似性计算将扫描全体项目空间, 计算量较大. 因此本文在 $C_u\cap C_v=\varnothing$ 时对 $sim(u,v)$ 不作计算.

2) 若 $C_u\cap C_v\neq\varnothing$, 则需要计算 $sim(u,v)$. 设

$C_t=C_u\cap C_v$, 抽取 I'_{uv} 中所有属于 C_t 所含任意项类的项目组成项集 I_t, 与 u, v 对 I_t 中所有项目的评分组成评分矩阵 $\boldsymbol{R}_t$, 然后基于领域最近邻对 $\boldsymbol{R}_t$ 中的未评分项进行评分预测.

定义 **1**. 领域最近邻. 设用户 u 和用户 v 的评分项类交集为 $C_t=\{c_1, c_2, \cdots, c_g\}$, 则对于 $\forall c_i\in C_t$ $(1\leqslant i\leqslant g)$, 择取用户-项目评分矩阵 $\boldsymbol{R}(m, n)$ 中所有属于 c_i 的项目评分及相应评分用户集 U_i 组成评分矩阵 $\boldsymbol{R}_i$, 然后基于 $\boldsymbol{R}_i$ 计算 u 与 $u_i\in U_i$ $(u\neq u_i)$ 的相似性 $sim(u, u_i)$, 则 $sim(u, u_i)$ 最大的前 K' 位用户称为 u 在 c_i 中的领域最近邻. 领域最近邻方法的基本思想有两点:

① 在实际应用中用户评分项很少, 且通常都根据其兴趣偏好集中在一个或少数几个项目类中, 因此目标用户在这些项目类中评分相对稠密, 同时由于相似性计算缩小到少数几个项目类中而非整个项目空间, 使算法计算量得到大幅减少;

② 商品项类实质上对应着用户的兴趣领域. 而两个用户在某个兴趣领域偏好相同, 未必在其他兴趣领域也相同, 因此不适合基于全体项目空间寻找其最近邻; 而在未评分项所属项类中寻找用户最近邻将更准确, 也更符合实际生活中的情况.

在寻找领域最近邻的过程中, 部分用户对目标用户的未评分项未作过评分, 因此不能对该项的推荐提供帮助, 在计算相似性时这些用户可以略去. 例如, 要预测表 3 中目标用户 u_1 对文学类项目 I_2 的评分(设为 5 分制评分), 需要在对文学类项目作过评分的用户集合 $\{u_2, u_3, u_4, u_5, u_6\}$ 中寻找 u_1 的文学类最近邻. 由于 u_3, u_4 对 I_2 没有评分, 故候选领域最近邻用户减为 $\{u_2, u_5, u_6\}$, 从而有利于降低相似性计算量.

Table 2　Omit Useless Users for Recommendation

表 **2**　略去对推荐无帮助的用户

User	Literature					
	I_1	I_2	I_3	I_4	I_5	I_6
u_1	2		5	1	5	2
u_2		5	2	1	1	
u_3	2		4	2	5	2
u_4	5			3		5
u_5	2	4	5	2	5	
u_6		5	1	3	4	3

设 u 在项类 c 中的领域最近邻集合为 $U_c=\{u_1, u_2, \cdots, u_{K'}\}$, 则对于评分矩阵 $\boldsymbol{R}_t$ 中的项目 $i\in c$ 且 $R_{u_k, i}=\varnothing$, u 对 i 的评分值 P_i 可由领域最近邻对 i 的评分进行加权逼近得到:

$$P_i=R_u+\frac{\sum_{u_k\in U_c} sim(u, u_k)\times(R_{u_k, i}-R_{u_k})}{\sum_{u_k\in U_c}(|sim(u, u_k)|)}, \tag{1}$$

式(1)中的 $sim(u, u_k)$ 表示用户 u 与用户 u_k 之间的相似性, $u_k\in U_c$ $(1\leqslant k\leqslant K')$, $R_{u_k, i}$ 表示 u_k 对 i 的评分, R_u 和 R_{u_k} 分别表示 u 和 u_k 在 c 中的平均评分.

通过领域最近邻方法完成对 R_t 中用户 u, v 未评分项的评分预测, 则 R_t 中任意项目 i 的评分 R_i 为

$$R_i=\begin{cases} r_i, & r_i\neq\varnothing, \\ P_i, & \text{otherwise}, \end{cases} \tag{2}$$

式(2)中 r_i 是来自 $R(m, n)$ 的原始评分, P_i 是采用领域最近邻方法得出的预测评分.

由于 R_t 中的所有项目都已有评分数据, 因此可以采用表 1 中的相似性度量方法计算 $sim(u, v)$. 类似地, 可计算出 u 与用户空间中其他用户的相似性, 然后取相似性从大到小排列的前 K 个用户作为 u 的最近邻集合 $U=\{u_1, u_2, \cdots, u_K\}$, $u\notin U$ 且 $sim(u, u_k)$ $(1\leqslant k\leqslant K)$ 由大到小排列, $sim(u, u_k)$ 为 u 与 u_k 的相似性.

2.3　推荐生成

在得到 u 的最近邻集合 U 后, 设 U 中各用户的评分项集合分别为 $I_1, I_2, \cdots, I_K$, I_u 为 u 的评分项集合, 令项目集合 $I_w=I_1\cup I_2\cup\cdots\cup I_K-I_u$, 则对于 $\forall i\in I_w$, $R_{u, i}=\varnothing$, 从而可采用式(2)预测 u 对 i 的评分值, 记为 $P_{u, i}$:

$$P_{u, i}=R_u+\frac{\sum_{u_k\in U} sim(u, u_k)\times(R_{u_k, i}-R_{u_k})}{\sum_{u_k\in U}(|sim(u, u_k)|)}, \tag{3}$$

式(3)中 $R_{u_k, i}$ 表示 u_k 对项目 i 的非空评分, R_{u_k} 表示 u_k 在与 u 的共同评分项集上的平均评分[12], R_u 则表示 u 在所有项目上的平均评分. 然后按 $P_{u, i}$ 值从大到小取前 N 个项目组成 top-N 推荐集 $I_{rec}=\{i_1, i_2, \cdots, i_N\}$ 并将其推荐给目标用户 u, 从而完成整个推荐过程.

算法 **1**. 基于领域最近邻的协同过滤推荐算法.

输入: 用户-项目评分矩阵 $\boldsymbol{R}(m, n)$、领域最近邻用户数 K'、最近邻用户数 K、推荐集 I_{rec} 项目数 N.

输出: 目标用户 u 的 top-N 推荐集 I_{rec}.

过程:

Step1. 设目标用户 u 和用户 v 的评分项集合分别为 I_u, I_v, 评分项所属项类集合分别为 C_u, C_v,

则 u 和 v 的评分项并集 $I'_{uv}=I_u\cup I_v$, 按照第 2.1 节的方法判别 v 是否为有推荐能力用户.

Step2. 若 v 属于有推荐能力用户, 则当 $C_u\cap C_v\neq\varnothing$ 时, 令 $C_t=C_u\cap C_v$, 抽取 I_{uv} 中所有属于 C_t 所含任意项类的项目组成项集 I_t, 与 u, v 对 I_t 中所有项目的评分组成评分矩阵 $\boldsymbol{R}_t$.

Step3. 按照定义 1 寻找 u 在 $\forall c_i\in C_t$ 中的领域最近邻.

Step4. 使用式(1)对 u 在 R_t 中未评分的 c_i 所属项目进行评分预测.

Step5. 循环执行 Step3 ~ Step4, 完成对 R_t 中用户 u 所有未评分项的评分预测, 类似地可完成对 R_t 中用户 v 所有未评分项的评分预测, 从而采用表 1 中的方法计算 u, v 的相似性 $sim(u,v)$.

Step6. 循环执行 Step1 ~ Step5, 得到 u 与其他用户的相似性, 取相似性从大到小排列的前 K 个用户作为 u 的最近邻集合 $U=\{u_1,u_2,\cdots,u_K\}$.

Step7. 对于 u 在最近邻评分项集合中的未评分项 i, 即 $R_{u,i}=\varnothing$, 采用式(3)预测 u 对 i 的评分 $P_{u,i}$.

Step8. 按 $P_{u,i}$ 值从大到小取前 N 个项目组成 top-N 推荐集 $I_{rec}=\{i_1,i_2,\cdots,i_N\}$ 并输出.

算法分析: 本文算法(记为 Proposed-CF)相对于文献[9]算法(记为 IRP-CF)在时间复杂度上的具体改进如下:

1) IRP-CF 基于全体项目空间搜寻未评分项 i 的相似邻居项目, 故搜寻时间复杂度为 $O(m\times n)$, 其中 m 表示用户总数, n 表示项目总数. Proposed-CF 由于只在 i 所属项类 c 中搜寻用户最近邻, 因此搜寻时间复杂度为 $O(m\times n_{ci})$, 其中 n_{ci} 表示对 i 做过评分的所有用户在 c 中的评分项总数. 若令 n_c 表示 c 中的项目总数, 则有 $n_{ci}\leqslant n_c\ll n$, 因此可得 $O(m\times n_{ci})\approx O(m)<O(m\times n)$.

2) Proposed-CF 不计算两种用户与目标用户之间的相似性, 一是无推荐能力用户, 二是在目标用户评分项所分布项类中无评分的用户, 从而进一步提高了用户最近邻的搜寻速度.

3 实验结果及分析

3.1 实验环境、数据集和度量标准

实验所用 PC 机的配置为 Intel Pentium 4 2.66GHz CPU; 1GB RAM; 操作系统为 Windows XP; 算法程序采用 PowerBuilder 9.0 实现; 数据库为 Access 2003.

MovieLens(http://movielens.umn.edu)是一个基于 Web 的研究型推荐系统, 通过用户对电影的评分(5 分制)进行电影推荐, 由美国明尼苏达大学开发并公布了两个评分数据集(http://www.grouplens.org/data/), 其中一个包含 943 位用户对 1682 部电影的 100000 条评分数据, 每位用户至少对 20 部电影进行了评分, 所有电影分属于 19 种电影类别; 另一个包含 6040 位用户对 3952 部电影的 1000209 条评分数据. 为了分析实验数据集的样本量成倍增加时对算法性能的影响, 本文从前一个数据集随机抽取 100, 200, 300 位用户的评分数据组成 3 个数据集, 分别记为 DS100, DS200, DS300, 表 3 和图 5 给出了这 3 个数据集各自的用户数量(quantity of users)、电影数量(quantity of movies)、评分数量(quantity of ratings)、稀疏等级(sparsity level)[3]及评分值(rating values)分布情况. 实验采用 All but one 协议将实验数据集每个用户的评分数据随机隐藏 1 个组成测试集, 然后基于其他评分数据(即训练集)来预测这些被隐藏评分.

Table 3 Experimental Datasets

表 3 实验数据集

Statistical Items	Datasets		
	DS100	DS200	DS300
Quantity of Users	100	200	300
Quantity of Movies	1292	1436	1486
Quantity of Ratings	10143	19255	29826
Sparsity Level	0.9215	0.9330	0.9331

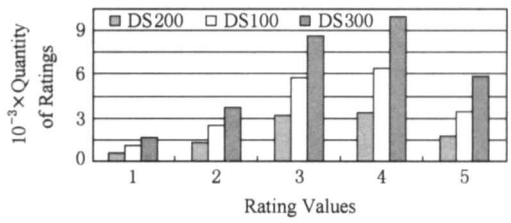

Fig. 5 Distribution of rating values.

图 5 评分值分布

评价推荐质量的度量标准主要有统计精度度量方法和决策支持精度度量方法两类[3]. 实验采用统计精度度量方法中广泛使用的平均绝对误差 MAE (mean absolute error):

设测试集中共有 H 条数据, 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

$$MAE=\frac{\sum_{i=1}^{H}|p_i-q_i|}{H},\qquad(4)$$

MAE 越小则表明评分预测越准确、推荐质量越高.

3.2　实验结果及分析

1) 算法 MAE 比较

本组实验基于稀疏等级最小的 DS100 数据集进行. 实验取领域最近邻用户数 K' 为 20, 用户相似性度量方法采用 Pearson 相关系数. 在最近邻用户数 K 分别取 4, 8, 12, 16, 20 时, 运行本文算法(Proposed-CF)和文献[9]算法(IRP-CF), 计算在不同最近邻用户数时 Proposed-CF 和 IRP-CF 各自的 MAE. 实验结果如图 5 所示:

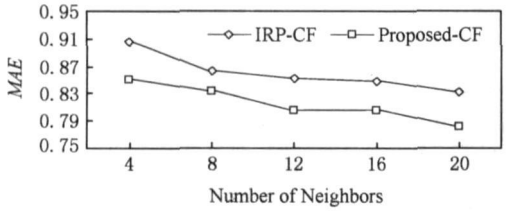

Fig. 6　Comparison of *MAE* of recommendation algorithms.

图 6　推荐算法的 *MAE* 比较

由图 6 可知, Proposed-CF 具有更小的 MAE. 这是由于用户可能存在多个兴趣领域, 且这些兴趣领域彼此之间并不必然相关, 因此在对未评分项进行评分预测时, 采用领域最近邻方法得到的最近邻与目标用户的兴趣偏好更为接近、对预测工作的帮助更大, 使得预测结果更准确, 从而提高了推荐质量. 而 IRP-CF 使用基于项目相似性的评分预测来填补用户评分项并集中的未评分值, 但该方法需要在两个用户之间的评分项目对集合上实施; 由于有时存在评分项目对集合为空的现象, 则相应的未评分值只得用 0 值填补, 这导致 IRP-CF 的 MAE 变大.

2) 数据集样本量成倍增加时对算法性能的影响

由图 6 可知, DS100, DS200, DS300 的评分数量分别为 10143, 19255, 29826, 其比例约为 1 ∶2 ∶3. 本组实验在这 3 个数据集上运行 Proposed-CF 并计算相应的 MAE. 实验结果如图 7 所示:

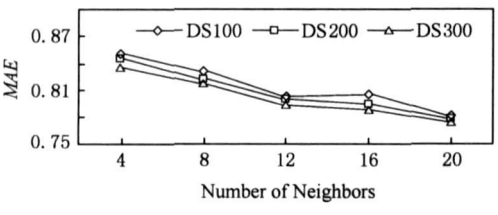

Fig. 7　Effect of datasets with various quantity of ratings.

图 7　数据集大小对算法的影响

从图 7 可以看出, Proposed-CF 在 DS300 上的 *MAE* 最小, 而在 DS100 上的 *MAE* 最大; 当最近邻数量增加到 20 时, 算法在 DS100, DS200, DS300 上的 *MAE* 最为接近. 由此可知, 当实验数据集的样本量成倍增加时对 Proposed-CF 算法性能的影响是良性的, 即 Proposed-CF 的推荐质量随样本量增加而得到小幅提高. 经分析认为, 这是由于电影类别数量相对固定, 因此当实验数据集变大后各个电影类别对应的用户及评分数量均随之增多, 使得 Proposed-CF 更容易找到相似性高的领域最近邻, 从而预测用户评分项并集中的空缺评分值时更准确, 最终使算法的 MAE 得到降低.

4　结　　论

基于项目评分预测的协同过滤推荐算法[9] 将用户评分项并集作为用户相似性计算基础, 但存在不必要的计算耗费. 本文进一步将用户评分项并集中的非目标用户区分为无推荐能力和有推荐能力两种类型, 对于无推荐能力用户不再计算其与目标用户的相似性, 从而提高算法效率和改善推荐实时性; 对于有推荐能力用户, 则在其与目标用户存在共同评分项类时采用"领域最近邻"方法对用户评分项并集中的未评分项进行评分预测, 从而使最近邻寻找更加准确. 实验结果表明, 本文提出的基于领域最近邻的协同过滤推荐算法能有效提高推荐质量. 下一步的研究工作将把基于 Rough 集理论的缺失值估算方法与本文算法进行结合, 以进一步提高算法推荐质量.

参　考　文　献

[1] Schafer J B, Konstan J A, Riedl J. E-commerce recommendation applications [J]. Data Mining and Knowledge Discovery, 2001, 5(1-2): 115-153

[2] Karypis G. Evaluation of item-based top-n recommendation algorithms[C] //Proc of the 10th Int Conf on Information and Knowledge Management. New York: ACM Press, 2001: 247-254

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

[4] Xing Chunxiao, Gao Fengrong, Zhan Sinan, *et al*. A collaborative filtering recommendation algorithm incorporated with user interest change[J]. Journal of Computer Research and Development, 2007, 44(2): 296-301 (in Chinese)
(刑春晓, 高凤荣, 战思南, 等. 适应用户兴趣变化的协同过滤推荐算法[J]. 计算机研究与发展, 2007, 44(2): 296-301)

[5] Sarwar B M, Karypis G, Konstan J A, *et al*. Application of dimensionality reduction in recommender system—A case study, TR 00-043 [R]. Minneapolis, USA: Department of Computer Science and Engineering, University of Minnesota, 2000

[6] Zhao Liang, Hu Naijing, Zhang Shouzhi. Algorithm design for personalization recommendation systems [J]. Journal of Computer Research and Development, 2002, 39(8): 986-991 (in Chinese)
(赵亮, 胡乃静, 张守志. 个性化推荐算法设计[J]. 计算机研究与发展, 2002, 39(8): 986-991)

[7] Zhou Junfeng, Tang Xian, Guo Jingfeng. An optimized collaborative filtering recommendation algorithm [J]. Journal of Computer Research and Development, 2004, 41(10): 1842-1847 (in Chinese)
(周军锋, 汤显, 郭景峰. 一种优化的协同过滤推荐算法[J]. 计算机研究与发展, 2004, 41(10): 1842-1847)

[8] Zhang Feng, Chang Huiyou. Employing BP neural networks to alleviate the sparsity issue in collaborative filtering recommendation algorithms [J]. Journal of Computer Research and Development, 2006, 43(4): 667-672 (in Chinese)
(张锋, 常会友. 使用BP神经网络缓解协同过滤推荐算法的稀疏性问题[J]. 计算机研究与发展, 2006, 43(4): 667-672)

[9] Deng Ailin, Zhu Yangyong, Shi Baile. A collaborative filtering recommendation algorithm based on item rating prediction [J]. Journal of Software, 2003, 14(9): 1621-1628 (in Chinese)
(邓爱林, 朱扬勇, 施伯乐. 基于项目评分预测的协同过滤推荐算法[J]. 软件学报, 2003, 14(9): 1621-1628)

[10] Ahn H J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem [J]. Information Sciences, 2008, 178(1): 37-51

[11] 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

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

Li Cong born in 1978. Ph. D. candidate in the School of Management, Hefei University of Technology. His main research interests include collaborative filtering, E-commerce and enterprise informatization.

李 聪, 1978年生, 博士研究生, 主要研究方向为协同过滤、电子商务、企业信息化.

Liang Changyong born in 1965. Received his Ph. D. degree from Harbin Institute of Technology in 2001. He is a professor and Ph. D. supervisor in the School of Management, Hefei University of Technology. His main research interests include collaborative filtering, intelligent decision support system, *etc*.

梁昌勇, 1965年生, 博士, 教授, 博士生导师, 主要研究方向为协同过滤、智能决策支持系统等.

Ma Li born in 1979. Received her M. S. degree from Chongqing Normal University in 2004. She is a lecturer in the Business College, China West Normal University. Her main research interests include management information system, software engineering, *etc*.

马 丽, 1979年生, 硕士, 讲师, 主要研究方向为管理信息系统、软件工程.

Research background

This work is supported by the National Natural Science Foundation of China under grant No. 70771037, the Key Project of Chinese Ministry of Education under grant No. 107067, and the Specialized Research Fund for the Doctoral Program of Higher Education of China under grant No. 20050359006.

With the rapid development of the Internet and E-commerce, customers are in urgent need of recommender systems to help them find right products quickly in E-commerce websites. Now the research and application of recommender systems are hot spots in the field of computer science and E-commerce. Many famous E-commerce websites have used recommender systems in their online applications, such as Amazon.com, eBay.com and dangdang.com. Currently collaborative filtering is the most successful and widely used recommendation technology in E-commerce recommender systems. However, there exist some problems in collaborative filtering algorithm: sparsity, real-time recommendation, cold-start and so on. How to solve these problems is the main research work for recommender systems. Some improved algorithms have been proposed by researchers, including item-based collaborative filtering and several model-based collaborative filtering (for instance, clustering and neural network technologies have been integrated with traditional collaborative filtering for improving the performance of algorithms). Our research aims to propose collaborative filtering algorithms with high performance and to implement available recommender systems for E-commerce websites.